

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



Sistema integrado e flexível para a gestão de processos de negócio e dos conteúdos associados

Nuno Ricardo d'Alte Curto

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major Automação

Orientador: José António Rodrigues Pereira de Faria (Prof. Dr.)

Julho de 2010

A Dissertação intitulada

**“SISTEMA INTEGRADO E FLEXÍVEL PARA A GESTÃO DE PROCESSOS DE NEGÓCIO E DOS
CONTEÚDOS ASSOCIADOS”**

foi aprovada em provas realizadas em 22/ Julho/2010

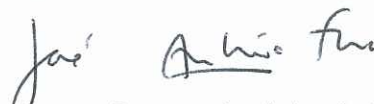
o júri



Presidente Professor Doutor Américo Lopes de Azevedo
Professor Associado do Departamento de Engenharia Industrial e Gestão
da Faculdade de Engenharia da Universidade do Porto



Professor Doutor Paulo Jorge Freitas Oliveira Novais
Professor Auxiliar do Departamento de Informática da Universidade do
Minho



Professor Doutor José António Rodrigues Pereira de Faria
Professor Auxiliar do Departamento de Engenharia Industrial e Gestão da
Faculdade de Engenharia da Universidade do Porto (Orientador)

O autor declara que a presente dissertação (ou relatório de projecto) é da
sua exclusiva autoria e foi escrita sem qualquer apoio externo não
explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros
extractos tomados de ou inspirados em trabalhos de outros autores, e
demais referências bibliográficas usadas, são correctamente citados.



Autor - Nuno Ricardo D'Alte Curto

Faculdade de Engenharia da Universidade do Porto

Resumo

Neste documento é apresentado um projecto desenvolvido no âmbito da disciplina Dissertação do Mestrado Integrado em Engenharia Electrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto (FEUP). O projecto teve por objectivo desenvolver dois módulos de software, um módulo para a gestão de processos de negócio e um módulo para a gestão de conteúdos. Estes módulos foram integrados num sistema de suporte à operação de sistemas de trabalho semi-estruturados, isto é, sistemas de trabalho que contém processos com diferentes níveis de estruturação, desde processos de *workflow* fortemente estruturados, até processos colaborativos pouco estruturados.

Numa primeira fase do trabalho, para se efectuar um levantamento de requisitos dos módulos a desenvolver, foi analisado como caso de estudo o sistema de trabalho dos Serviços Técnicos e de Manutenção (STM) da FEUP, o qual inclui um vasto conjunto de processos com diferentes níveis de estruturação e graus de complexidade, desde processos elementares, p.ex., manutenção preventiva de um equipamento, até processos complexos, p.ex., empreitada correspondente a um novo edifício. Para além dos diferentes níveis de estruturação dos processos, outra característica importante destes sistemas é o facto dos processos produzirem múltiplos documentos e envolverem múltiplas interacções com entidades externas, em particular fornecedores e prestadores de serviços.

Os sistemas de suporte tradicionais, como os sistemas de gestão de workflow, não são adequadas para suportar este tipo de sistemas de trabalho, porque não oferecem o grau de flexibilidade necessário para gerir processos e conteúdos com diferentes níveis de estruturação. Assim, com base na análise de requisitos efectuada, foi desenvolvido um gestor de processos, que com o auxílio de uma *checklist* permite a gestão dos mesmos de uma forma flexível, dentro de um conjunto de regras, e um gestor de conteúdos que permite agregar conteúdos provenientes de diferentes contextos e de diferentes suportes (emails, ficheiros, base de dados, etc.), apresentando-os sob a forma de uma árvore.

A implementação dos módulos foi efectuada em C# e .NET. Na camada de acesso a dados foi utilizada a tecnologia LINQ que oferece uma interface orientada a objectos para aceder aos dados. Após a implementação, os módulos foram integrados como componentes (Peças Web) na aplicação de suporte aos STM, baseada em Microsoft SharePoint.

Abstract

In this document it's presented a project developed for the Dissertation Thesis of the Master in Electrical and Computers Engineering, Faculty of Engineering, University of Porto (FEUP). The project aimed to develop two software modules: a module for managing business processes and a module for content management. These modules were integrated into an operation support system of semi-structured working systems, i.e., working systems that contains processes with different levels of organization, from the highly structured workflow processes, to poorly structured collaborative processes.

In the first phase of work, to get the requirements of the modules to develop, was examined as a case study the working system of Technical Services and Maintenance (STM) of FEUP, which includes a broad set of processes with different levels of structure and degrees of complexity, from the elementary processes, e.g., preventive maintenance of an equipment, even complex processes, e.g., contract corresponding to a new building. Apart from the different levels of structural processes, another important characteristic is that the processes produce multiple documents and involve multiple interactions with external entities, in particular suppliers and services.

The traditional support systems, such as workflow management systems, are not adequate to support this kind of working systems, because they don't offer the degree of flexibility needed to manage content and processes with different levels of structuring. Thus, based on the requirements analysis conducted, was developed a process manager that, with the aid of a checklist, allows their management in a flexible way within a set of rules, and a content management module that allows you to aggregate content from different contexts and different media (emails, files, databases, etc..), presenting them in the form of a tree.

The implementation of the modules was done in C# and .NET. In the data access layer was used the LINQ technology that offers object-oriented interface for accessing data. After implementation, the modules were integrated as components (Web Parts) on support application to STM, based on Microsoft SharePoint.

Agradecimentos

A apenas algumas horas de concluir a presente dissertação, estou a pensar na longa caminhada que realizei até aqui e com uma profunda nostalgia encontro pessoas que foram verdadeiramente fundamentais neste percurso. Pretendo portanto prestar-lhes os meus agradecimentos neste espaço.

Em primeiro lugar quero agradecer aos meus pais e irmã, que sempre depositaram em mim uma confiança inabalável.

Quero agradecer à minha namorada, Joana Miranda, pelo equilíbrio, apoio e motivação que me transmitiu.

Ao meu amigo e colega de curso, João Silva, que para além de ter efectuado uma revisão deste documento, tornou-se um amigo para toda a vida, fruto das milhares de horas que estivemos juntos por motivos académicos.

Quero ainda agradecer às pessoas que me apoiaram ao longo da realização deste projecto. Ao meu orientador, Prof. José Faria, pelo conhecimento que me transmitiu e sobretudo pela forma como o fez e aos Engenheiros Miguel Silva e Helder Lourenço, pelo apoio técnico que me deram, nomeadamente ao nível de Microsoft SharePoint, .NET, ASP.NET e C#.

A todos, um muito obrigado.

Conteúdo

1	Introdução	1
1.1	Objectivos e Motivação	1
1.2	Metodologia	2
1.3	Organização do Documento	3
2	Fundamentos sobre sistemas de trabalho	5
2.1	Conceito de processo de negócio	5
2.1.1	Processos de Workflow	8
2.1.2	Processos Colaborativos	9
2.1.3	Processos Semi-Estruturados	11
2.2	Sistemas de trabalho semi-estruturados	16
2.3	Gestão de Conteúdos	19
2.4	Síntese	20
3	Análise do sistema de trabalho	21
3.1	Organização e actividades	21
3.2	Análise de processos	27
3.2.1	Processo Executar intervenção de manutenção	27
3.2.2	Processo Realizar projecto/empreitada	30
3.3	Análise das necessidades	32
4	Gestor de processos	35
4.1	Análise de requisitos	35
4.2	Concepção	37
4.2.1	Arquitectura	37
4.2.2	Modelo de dados	39
4.2.3	Interfaces	47
4.3	Implementação	50
4.3.1	Acesso a Dados	50
4.3.2	Interfaces	53
5	Gestor de conteúdos	59
5.1	Análise de requisitos	59
5.2	Concepção	60
5.2.1	Arquitectura	60
5.2.2	Modelo de dados	63
5.2.3	Interface	64
5.3	Implementação	66

6	Conclusões e perspectivas de desenvolvimento	71
A	Lista de equipamentos, sistemas e infraestruturas dos STM	73
	Referências	77

Lista de Figuras

1.1	Fases do desenvolvimento da dissertação	2
2.1	Processo de negócio	7
2.2	Localização dos diferentes tipos de processos na gama de especificação	8
2.3	Esquema conceptual de um processo de <i>workflow</i>	8
2.4	Gestão de um processo exemplo num ERP ou WFMS	9
2.5	Esquema conceptual de um processo colaborativo	10
2.6	Esquema conceptual de um processo semi-estruturado	11
2.7	Exemplo de uma checklist partilhada num protótipo de uma aplicação UAM	13
2.8	Criação de modelo através do DECLARE Designer (fonte: [1])	16
2.9	Execução de processo através do DECLARE Worklist (fonte: [1])	17
2.10	Constituição de um sistema de trabalho semi-estruturado e sua visão de conjunto	18
2.11	Gestão de conteúdos num WFMS	19
2.12	Gestão de conteúdos num sistema de trabalho semi-estruturado	20
3.1	Área de actuação dos STM	22
3.2	Estrutura organizacional dos STM	22
3.3	Processo Executar intervenção de manutenção	29
3.4	Processo Realizar projecto/empreitada	31
4.1	Arquitectura geral do gestor de processos	38
4.2	Modelo de dados do pacote modelos dos processos	41
4.3	Exemplo de atributo a associar a uma tarefa	41
4.4	Modelo de dados do pacote instâncias dos processos	42
4.5	Organização hierarquizada dos STM	43
4.6	Modelo de dados do pacote navegação	44
4.7	Modelo de dados completo	46
4.8	Esboço da interface de criação de instâncias de processos	47
4.9	Esboço da interface de listagem de instâncias de processos numa área de actividade	48
4.10	Esboço da interface de edição de instâncias de processos	49
4.11	Criação da lista de empregados usando uma abordagem tradicional em C#	51
4.12	Criação da lista de empregados usando LINQ	51
4.13	Função que retorna uma instância de processo usando LINQ para SQL	52
4.14	Função que permite fazer <i>update</i> a uma instância de processo usando LINQ para SQL	53
4.15	Função que permite apagar uma instância de processo usando LINQ para SQL . .	53
4.16	Interface de criação de instâncias de processos	54
4.17	Interface de listagem de processos	55
4.18	Interface de edição de instâncias de processos	56

4.19	Popup com informação adicional	57
5.1	Arquitectura geral do gestor de conteúdos	61
5.2	Modelo de dados do pacote árvore de conteúdos	63
5.3	Modelo de dados do gestor de conteúdos	65
5.4	Esboço da interface do gestor de conteúdos	66
5.5	Dados da tabela tipoNo	67
5.6	Dados da tabela NoConteudo	67
5.7	Interface do gestor de conteúdos	68
5.8	Explorador do windows para organização dos conteúdos	69

Lista de Tabelas

3.1	Âmbito de actuação das unidades pertencentes aos STM	24
3.2	Principais actividades internas dos STM	25
3.3	Empresas responsáveis pela manutenção dos equipamentos dos STM	26
4.1	Requisitos do módulo de gestão flexível de processos	36
5.1	Requisitos do módulo de gestão de conteúdos	60
A.1	Equipamentos, sistemas e infraestruturas no âmbito das instalações eléctricas . .	73
A.2	Equipamentos, sistemas e infraestruturas no âmbito dos sistemas de alimentação de emergência	73
A.3	Equipamentos, sistemas e infraestruturas no âmbito da gestão técnica centralizada	74
A.4	Equipamentos, sistemas e infraestruturas no âmbito do AVAC	74
A.5	Outros equipamentos, sistemas e infraestruturas	75

Abreviaturas e Símbolos

BD	Base de dados
C#	C Sharp
ERP	Enterprise Resource Planning
FEUP	Faculdade de Engenharia da Universidade do Porto
LINQ	Language Integrated Query
MCRE	Manutenção curativa com base em recursos externos
MRI	Manutenção com base em recursos internos
MPRE	Manutenção programada com base em recursos externos
SI	Sistema de informação
SP	SharePoint
STM	Serviços Técnicos e de Manutenção
UAM	Unified Activity Management
URL	Uniform Resource Locator
WFMS	Workflow Management Systems

Capítulo 1

Introdução

Neste capítulo efectua-se uma breve contextualização do projecto desenvolvido, começando por definir os seus objectivos e apontando as motivações e as necessidades que levaram à sua criação. Será, ainda, apresentada a metodologia seguida ao longo do projecto e organização do presente documento.

1.1 Objectivos e Motivação

Este projecto tem por objectivo a análise de requisitos, a especificação e o desenvolvimento de um sistema de gestão de processos e da documentação associada, que permita gerir processos com diferentes níveis de estruturação. O sistema a desenvolver tem como aplicação de referência o sistema de trabalho dos Serviços Técnicos e de Manutenção (STM) da Faculdade de Engenharia da Universidade do Porto (FEUP).

Este sistema de trabalho, pela diversidade dos processos que engloba, desde as manutenções até aos projectos, é representativo de um vasto conjunto de sistemas com características idênticas existentes noutras organizações. De facto, nos STM são realizados processos com características e domínios completamente diferentes, o que dificulta a tarefa de quem está responsável pela parte de concepção do Sistema de Informação (SI). Se por um lado alguns processos podem ser modelados rigidamente recorrendo às técnicas comuns de workflow (p.ex: swimlanes), outros não são propícios a tal abordagem, pelo facto de as suas características e execuções possuírem uma elevada variabilidade associada às circunstâncias em que o processo se desenrola ou, até mesmo, por depender da experiência e das decisões efectuadas pelos actores responsáveis pela sua execução.

No passado, foi testada nos STM a aplicação de um sistema convencional de gestão da manutenção, que se revelou inadequado às características do serviço, por não oferecer fundamentalmente um grau de flexibilidade capaz de cobrir a diversidade dos processos e da documentação gerida pelos STM. Se por um lado, um sistema de suporte rígido e muito

estruturado, como é o caso dos sistema de workflow, não é o ideal para estes sistemas de trabalho, a solução também não passa por um sistema não estruturado, baseado em ferramentas de produtividade pessoal. Portanto, é necessário encontrar-se uma solução que seja simultaneamente flexível e capaz de se adaptar às necessidades dos diferentes tipos de processos, mas por outro lado, que permita estruturar os processos de forma a ser possível ter um controlo da sua execução. Foi com esta ideia em mente que foram especificados, estabelecidos e definidos os objectivos deste projecto, que passam pelo desenvolvimento de dois módulos, um para a gestão de processos de negócio e outro para a gestão dos conteúdos associados. Após o desenvolvimento destes módulos, pretende-se integrá-los na aplicação de suporte aos STM, baseada em Microsoft SharePoint.

1.2 Metodologia

Para a concretização dos objectivos propostos, esta dissertação foi dividida em duas componentes: concepção e implementação. A Figura 1.1 posiciona cada uma das fases do projecto na componente em que se realizou e ilustra a interação entre as diferentes fases e suas precedências.

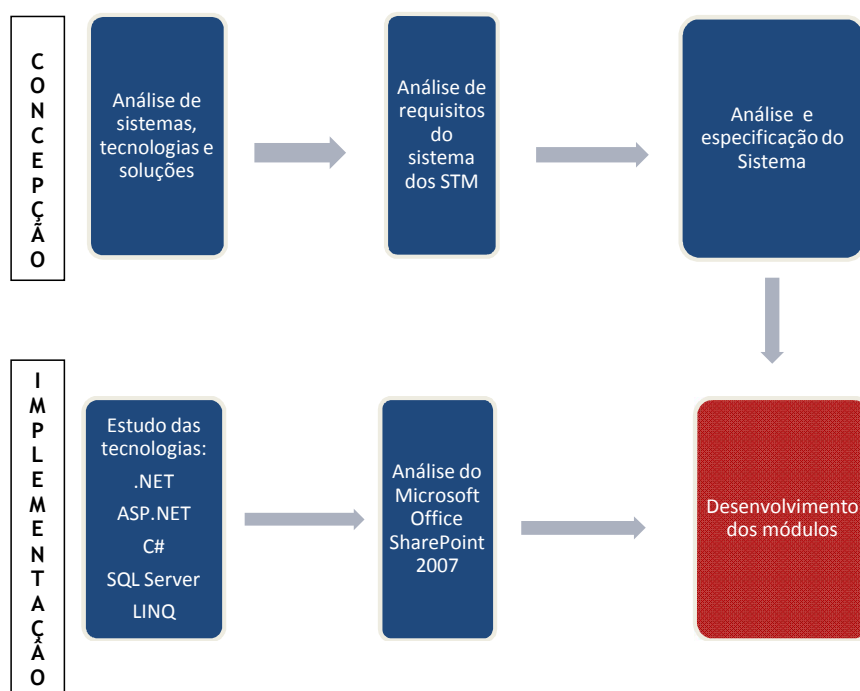


Figura 1.1: Fases do desenvolvimento da dissertação

Como é possível constatar através da Figura 1.1, para chegar à fase final do projecto (rectângulo a vermelho) e ao seu objectivo final, decidiu-se desenvolver em paralelo a componente de concepção e implementação. Não seria possível chegar à última fase "Desenvolvimento dos módulos" sem ser efectuado um trabalho de concepção prévio. Contudo, era possível fazer-se o

trabalho de concepção e só de seguida começar-se o de implementação, através do "Estudo das Tecnologias: .NET, ASP.NET, C#, SQL Server e LINQ", mas tal abordagem não foi adoptada por considerar-se que seria mais benéfico e eficiente efectuar-se a parte de estudo tecnológico, que iria servir de suporte à fase de implementação dos módulos para o SharePoint, em paralelo com a componente de concepção.

Na componente de concepção começou-se por fazer uma análise de sistemas, tecnologias e soluções, tendo como base artigos científicos e livros, que serviram de suporte à realização do trabalho e permitiram a formulação e consolidação de uma opinião em relação aos assuntos abordados nesta dissertação, terminando esta fase com a elaboração do Capítulo 2. De seguida, analisou-se os STM da FEUP, o que originou um conjunto de requisitos que o sistema a desenvolver teria de satisfazer. Na última fase desta componente especificou-se os módulos do SI que foram implementados, tendo como base os paradigmas de gestão de processos semi-estruturados, estudados na primeira fase, e as necessidades do sistema, identificadas na segunda fase.

Na componente de implementação começou-se, numa primeira fase, por estudar ferramentas de programação que permitiram a implementação dos módulos em SharePoint, através de [2, 3, 4, 5, 6, 7]. Na fase seguinte, efectuou-se um estudo de SharePoint com base em [8, 9], que incidiu essencialmente no domínio da ferramenta na óptica do utilizador e na aprendizagem de alguns pormenores relativos à programação em SharePoint. Por ultimo, após adquirir todos os conhecimentos técnicos e efectuar-se toda a componente de concepção, foram desenvolvidos os dois módulos especificados previamente.

1.3 Organização do Documento

Esta dissertação encontra-se dividida em seis capítulos, dos quais o primeiro é composto por esta introdução ao projecto.

No capítulo seguinte, Capítulo 2, são apresentados os fundamentos que serviram de base à realização deste trabalho, nomeadamente ao nível dos diferentes tipos de processos e sistemas de suporte.

No Capítulo 3, pretende-se estudar e extrair as características fundamentais do sistema de trabalho semi-estruturado, com o objectivo de encontrar um conjunto de requisitos que um sistema de suporte a este sistema de trabalho deva satisfazer. Para tal, utilizar-se-á como caso de estudo os STM da FEUP.

No Capítulo 4, após se ter identificado no capítulo anterior os requisitos que um sistema de suporte deveria satisfazer ao nível da gestão de processos, aborda-se aspectos de concepção e implementação relativos ao módulo gestor de processos.

No Capítulo 5, após se ter identificado no Capítulo 3 os requisitos que um sistemas de suporte deveria satisfazer ao nível da gestão de conteúdos, aborda-se aspectos de concepção e implementação relativos ao módulo gestor de conteúdos.

No último capítulo, Capítulo 6, apresentam-se as conclusões do trabalho desenvolvido e são apontadas perspectivas para futuros desenvolvimentos.

Capítulo 2

Fundamentos sobre sistemas de trabalho

Este capítulo tem como objectivo contextualizar os temas abordados nesta dissertação, dando ênfase aos processos semi-estruturados por serem tema central deste trabalho.

Assim, começa com uma introdução, na secção 2.1, ao conceito de processo de negócio, assim como aos seus diferentes tipos e sistemas de suporte.

Posteriormente, na Secção 2.2 é efectuada uma introdução ao conceito de sistema de trabalho semi-estruturado.

Na secção 2.3 são apresentadas as características requeridas a um sistema de gestão de conteúdos de um sistema de trabalho semi-estruturado.

Por último, em 2.4, é realizado um resumo sucinto dos temas abordados neste capítulo.

2.1 Conceito de processo de negócio

Actualmente as organizações são confrontadas com diversos problemas de ordem económica e tecnológica que têm um impacto directo na forma como realizam as suas actividades.

Se uma empresa pretende ser bem sucedida no mercado tem que ser competitiva, para atingir esse objectivo temos vindo a assistir as empresas a aderirem ao uso dos processos de negócio (Hammer, 2001).

As tecnologias existentes e em especial a Internet, alteram drasticamente a forma como as empresas comunicam com os seus clientes. Canais de comunicação mais eficientes possibilitam o aumento de vendas e de serviços prestados. Como resultado, as empresas são confrontadas com um aumento significativo das suas actividades [10].

Para se tornarem mais competitivas e para dar resposta ao aumento das actividades, as organizações procuram formas de aumentar a eficiência dos seus processos de negócio internos. Contudo, existem vários factores que tornam complexo o controlo de processos de forma eficiente:

- Um processo de negócio pode-se alongar por um longo período de tempo, podendo levar anos a ser finalizado.
- Um processo de negócio pode envolver pessoas que pertencem a diferentes unidades da estrutura da organização.
- A mesma pessoa pode ser envolvida em muitos processos de negócio simultaneamente.

Aliado às dificuldades impostas por estes factores está o facto de, tipicamente, estarem em curso simultaneamente um elevado número de processos, o que não permite a nomeação de um responsável que supervisione a execução de cada processo separadamente.

A mudança de paradigmas de trabalho tem afectado a forma como se gerem as actividades nas empresas, o que se reflecte numa procura cada vez maior de tecnologias de informação. Em [10] é partilhada a mesma opinião, ao ser afirmado que a solução consiste em transferir algumas funções de coordenação e controlo de processos para um sistema informático de suporte. Devido a este facto, muitas organizações são forçadas a criar ou reestruturar os seus processos, de forma a procurar não somente a eficiência na execução dos seus processos, mas também a eficácia, com o objectivo de executar as suas tarefas da melhor forma possível. Atinge-se, assim, um grau de satisfação muito superior por parte dos clientes, devido aos produtos e serviços daí resultantes.

Como os processos de negócio não possuem todos as mesmas características e os mesmos graus de formalidade, os sistemas de suporte necessitem de ser escolhidos de acordo com as características dos processos, para que possam ser, efectivamente, extraídos benefícios na utilização destes sistemas.

A utilização de sistemas de apoio à gestão de processos exige a total compreensão do conceito processo de negócio. Este conceito é algo que tem as suas raízes históricas em diferentes perspectivas e nas quais é difícil precisar o marco do seu nascimento. Se recuarmos no tempo e pensarmos no trabalho elaborado por um artesão, é possível concluir que este já desempenhava uma sequência de actividades desde a recolha da matéria-prima, passando por adquirir as medidas do cliente, pelo corte e costura e terminando com a facturação e contabilização do trabalho efectuado. Embora todas estas actividades realizadas sequencialmente pelo artesão pudessem ser consideradas apenas um único processo, ele quando as realizava certamente que não estaria a pensar neste conceito, porque seguramente nem sequer tinha consciência da sua existência.

No início dos anos 90, um processo era designado como "um conjunto de actividades estruturadas e bem definidas, criadas para produzir uma saída para o cliente". Este conceito mudou ligeiramente em 1993 quando Hammer & Champy afirmaram que um processo consistia "numa colecção de tarefas que transformam entradas e as converte em saídas criando valor para o cliente". Esta definição está mais perto do que actualmente entendemos como processo e introduz a ideia de fluxo. A Figura 2.1 ilustra precisamente este conceito.

Os processos de negócio coordenam os seus elementos: pessoas, recursos, sistemas e trabalho. Sem processos de negócio tudo seria concluído de uma forma *ad-hoc* ou, provavelmente, descoordenada [11]. O objectivo dos processo é sempre o mesmo: transformar entradas em

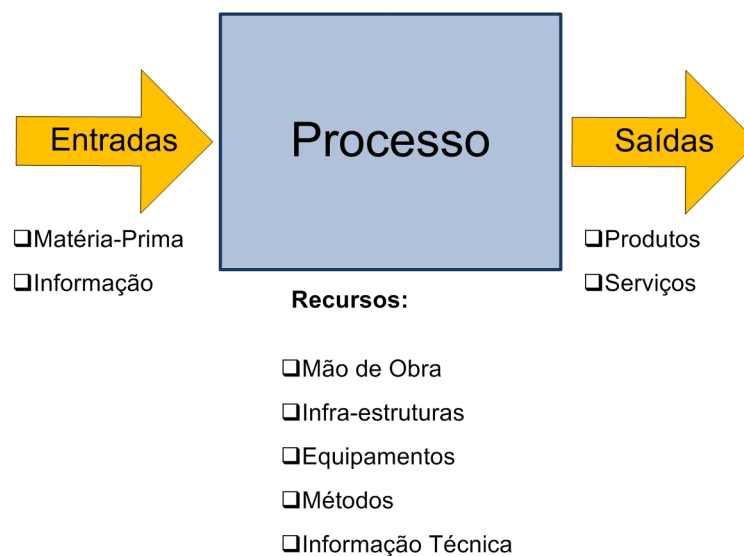


Figura 2.1: Processo de negócio

saídas. Contudo, a forma como este objectivo é alcançado varia de processo para processo e está intrinsecamente relacionado com as características do trabalho a realizar. Enquanto que o processo realizado pelo artesão era composto por um conjunto de tarefas repetitivas e pouco variáveis do ponto de vista da sua execução, parte dos processos realizados actualmente em muitas organizações não partilham tais características. Tratam-se de processos que não possuem uma ordem rígida entre as suas tarefas (fluxo variável) o que os torna difícil de gerir e modelar. Isto torna evidente a existência de processos com diferentes níveis de estruturação e de especificação. Assim, é possível identificar-se uma gama de especificação de processos, que varia entre processos pouco estruturados/especificados e processos muito estruturados/especificados. A Figura 2.2 ilustra a gama de especificação e o posicionamento dos diferentes tipos de processos (processos de workflow, processos semi-estruturados e processos colaborativos) nessa mesma gama. A diferença entre estes três tipos de processos reside na forma como o trabalho é estruturado para se transformar as entradas do processo nas saídas.

É frequente assistir-se a uma evolução do processo de negócio no que diz respeito à gama de especificação, podendo esta variar para um determinado processo ao longo do seu ciclo de vida. Bernstein e Schucan [12], por exemplo, fornecem uma descrição de como o processo de uma transferência bancária de dinheiro evoluiu no tempo. Antes de estarem estabelecidas formalmente todas as regras inerentes a este processo no sector bancário, começou por ser um processo muito pouco estruturado que envolvia o envio por correio de uma carta *ad-hoc* (estrutura da carta era variável em cada processo). Com a maturação da indústria bancária a especificação deste processo cresceu significativamente. Actualmente, uma transferência bancária de dinheiro trata-se de um processo extremamente estruturado e rígido, com um conjunto fixo de atributos a si associados. Este processo mostra que a experiência adquirida na gestão do processo ao longo do seu ciclo de vida, faz com que a sua execução vá sendo aperfeiçoada até se atingir um padrão de execução.



Figura 2.2: Localização dos diferentes tipos de processos na gama de especificação

Embora os processos existam em todas as organizações pode não ser fácil identificá-los, e normalmente não o é. É importante adoptar uma definição que dê uma ideia clara sobre o que deve, ou não, ser considerado um processo. Nesse sentido, nas secções seguintes será feita uma descrição mais detalhada de cada tipo de processo mencionado anteriormente.

2.1.1 Processos de Workflow

Um processo de workflow consiste num processo de negócio que possui todas as suas tarefas identificadas e definidas, estando inclusivamente pré-definida a ordem pela qual devem ser realizadas. Tal como ilustra a Figura 2.3 nestes processos estão identificados todos os caminhos que o processo poderá seguir.

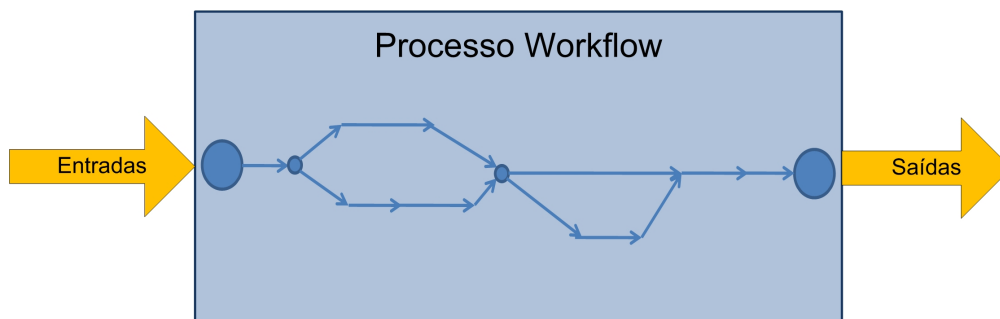


Figura 2.3: Esquema conceptual de um processo de *workflow*

Esta abordagem só é viável para processos de negócio que estejam sujeitos a pouca variabilidade, como é o caso do processo de transferência bancária de dinheiro mencionado anteriormente, uma vez que consiste num processo em que todas as suas tarefas estão identificadas e o seu fluxo também, podendo-se assim especificar todos os estados possíveis que poderão ocorrer e consequentemente todos os rumos que o seu fluxo poderá seguir.

2.1.1.1 Sistemas de suporte

Os sistemas de suporte a processos de negócio mais comuns são os Workflow Management Systems (WFMS). Estes sistemas são vocacionados para a gestão de processos com um modelo

rigidamente definido e como nesse modelo estão contidas todas as tarefas do processo e as suas precedências, estes sistemas têm a capacidade para criar e atribuir automaticamente as tarefas aos actores ao longo da execução do processo. Por exemplo, no processo presente na Figura 2.4 o sistema começaria por criar a Tarefa A e iria notificar o Actor A a necessidade da sua realização. Após o Actor A realizar a Tarefa A, o sistema iria proceder da mesma forma (criar a tarefa e notificar), mas agora para o Actor B. Quando o Actor B terminasse a Tarefa B o procedimento anterior repetiria-se novamente e o trabalho seria então encaminhado de novo para o Actor A, para que ele realizasse a Tarefa C. Este exemplo mostra que para além destes sistemas atribuírem as tarefas aos actores ao longo do ciclo de vida do processo, também controlam todas as interacções entre os intervenientes no processo.

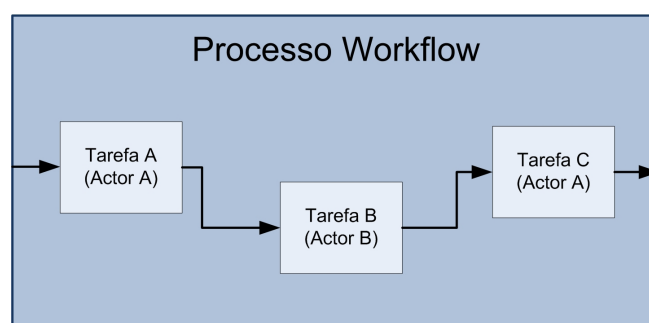


Figura 2.4: Gestão de um processo exemplo num ERP ou WFMS

Sistemas de gestão de processos com estas características têm um carácter prescritivo no sentido em que o sistema é quem informa ao utilizador as tarefas que deve realizar em cada momento, fruto da informação que contém relativa ao modelo do processo.

2.1.2 Processos Colaborativos

No extremo oposto aos processos de workflow, na gama de especificação de processos, estão os processos colaborativos. São processos em que os seus actores para transformar entradas em saídas têm, inicialmente, pouca ou nenhuma informação que os auxilie na forma como o poderão fazer. Dadas as características destes processos é normal não possuírem modelo, o que tem impacto na forma como são geridos ao longo do seu ciclo de vida. Na ausência de um modelo, para gerir a execução destes processos os actores não têm nenhuma referência, o que os obriga a ir definindo um rumo para o processo à medida que vão obtendo informações pertinentes. A Figura 2.5 ilustra a existência de um conjunto de rumos e tarefas em número indeterminado, dependendo dos actores responsáveis pela realização do processo e da forma como for decorrendo a execução do mesmo. Este tipo de processos envolvem, tipicamente, incerteza e decisões por parte dos actores, estando por isso regularmente associados a processos do conhecimento.

Um exemplo típico de um processo colaborativo é um trabalho académico desenvolvido por um grupo de alunos. No início do trabalho não é possível especificar totalmente quais serão as fases ou as tarefas do trabalho, nem definir precedências entre elas. Mas com o desenrolar

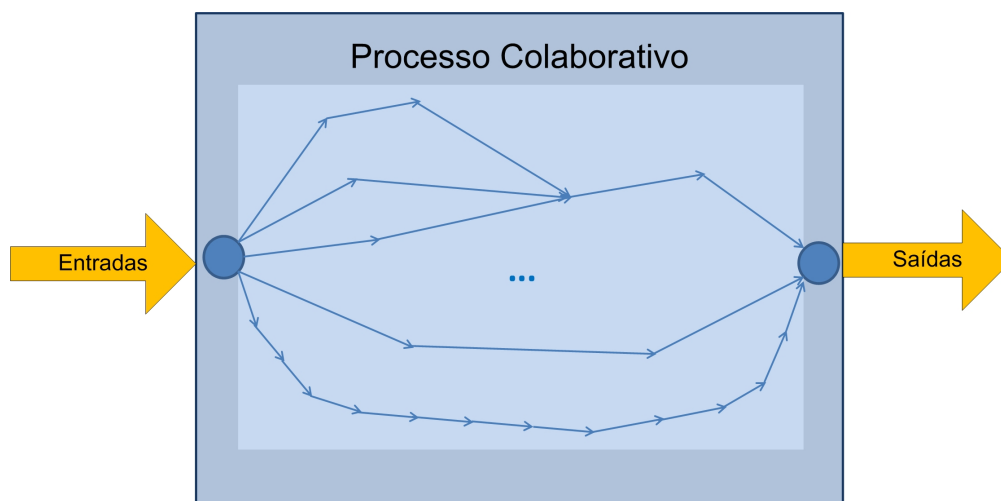


Figura 2.5: Esquema conceptual de um processo colaborativo

do trabalho certamente que irão surgindo tarefas a atribuir a cada elemento do grupo e uma sequência lógica das diferentes fases do trabalho e das tarefas a desenvolver. Este tipo de trabalho dificilmente poderia ser um processo de workflow, porque está associado a este processo uma elevada variabilidade nas suas tarefas, ou seja, o trabalho da disciplina A certamente que será bastante diferente do da disciplina B.

2.1.2.1 Sistemas de suporte

Ferramentas como o Google Groups, o Google Sites e o Lotus Notes são sistemas de suporte a processos colaborativos. São sistemas extremamente flexíveis, que fornecem um conjunto de componentes (para tarefas, notícias, gestão de ficheiros, etc...) pré-definidos que facilitam o desempenho das tarefas aos seus utilizadores. Embora forneçam todos estes recursos, o sistema não dá indicações relativamente à forma como eles devem ser utilizados, ficando isso ao critério do utilizador aquando a gestão do processo.

Estes sistemas são os ideais para processos colaborativos pela sua flexibilidade. Como quando se inicia um processo colaborativo não há um modelo pré-definido, o utilizador terá de ter à sua disposição um sistema que lhe permita com bastante flexibilidade gerir o processo em função das suas necessidades. Assim, em vez de se tratarem de sistemas prescritivos como os ERPs ou WFMSs, em que o sistema é quem decide aquilo que cada um tem de fazer e quando o tem de fazer, nestes sistemas os actores do processo é que são responsáveis por essas tarefas.

Quando se cria um site no Google Groups o sistema inicialmente não contém qualquer tipo de estruturação. Não diz quais serão as tarefas do projecto, a estrutura documental necessária, nem contém regras associadas ao processo. Mas com o desenrolar do processo o utilizador poderá moldar o sistema às suas necessidades, embora nunca o consiga transformar num sistema prescritivo, mas sim num sistema de apoio à execução do processo.

2.1.3 Processos Semi-Estruturados

Segundo [13], os maiores ganhos de produtividade ao nível dos processos de negócio foram alcançados através da formalização dos processos em WFMS. Contudo, muitos processos não são propícios a esta abordagem. Foi neste contexto que a IBM, em [13], introduziu o conceito de *artful process*, ou processo semi-estruturado.

Um processo semi-estruturado consiste num processo com uma estrutura que pode ser estável a um nível abstracto, mas ao nível dos seus detalhes não. Está dependente dos atributos, experiência e julgamentos dos actores responsáveis pela sua execução. Estes processos são também denominados por *artful*, pelo facto de à forma como são executados estar associada uma "arte" que não pode ser facilmente dissociada dos seus actores.

O facto de terem uma grande variabilidade e estarem muito dependentes dos seus executantes, não impossibilita a criação de um modelo. Um modelo de um processo semi-estruturado pode conter restrições obrigatórias e restrições opcionais (políticas, boas práticas de execução, etc...). Sendo assim, aquando a execução do processo os actores têm total liberdade para conduzir o processo da forma que considerarem mais conveniente, desde que não violem as restrições obrigatórias e procurem seguir as opcionais. Isto faz com que possa haver diferentes execuções do mesmo processo (Figura 2.6).

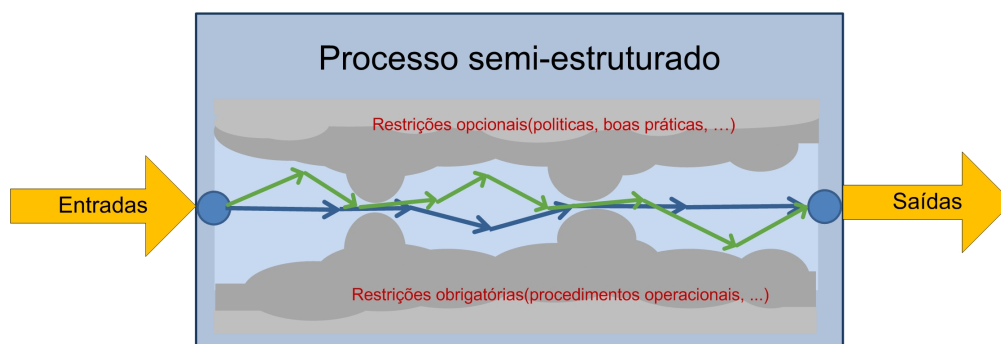


Figura 2.6: Esquema conceptual de um processo semi-estruturado

Como exemplo, podemos considerar o processo de acompanhamento de um processo judicial por parte de um advogado. É óbvio que, neste caso, a forma como o processo irá ser conduzido por parte do advogado depende dos seus conhecimentos (atributos), do seu traquejo (experiência) e das suas decisões (julgamentos) ao longo da realização do processo. O mesmo processo judicial certamente que seria conduzido de formas diferentes caso fosse entregue a dois advogados distintos. Contudo, isto não implica que este processo não possa ser especificado com um grau de formalidade que permita definir directrizes quanto à sua execução, ou seja, embora a execução do processo judicial possa ser variável de advogado para advogado, é possível encontrar um conjunto de procedimentos operacionais, fases do processo, tarefas obrigatórias, boas práticas e políticas que podem fazer parte do modelo deste processo e que serviam de referência a futuras execuções, sem que condicione a flexibilidade necessária em processos deste tipo. Os dois rumos (verde e

azul) presentes na Figura 2.6 poderiam ser os rumos seguidos por os dois advogados na execução do mesmo processo judicial, desde que eles não infringissem as restrições do modelo do processo.

2.1.3.1 Sistemas de suporte

Os sistemas de suporte de processos mais comuns (WFMS) estão por norma focados no suporte de processos organizacionais rígidos [14]. Uma das maiores fontes de inflexibilidade dos WFMS é a sua interpretação literal do termo *workflow*. Este termo sugere uma visão dos processos de negócio baseada no fluxo de trabalho. Esta visão tem bons resultados para processos repetitivos, mas não se adequa a processos mais variáveis como os processos que envolvem conhecimento [15], onde é exigido um grau de flexibilidade superior. Isto é o que se passa com os processos semi-estruturados. Se por um lado sistemas de suporte a processos de workflow (p.ex. WFMSs) são muito rígidos para os suportar, os sistemas de suporte a processos colaborativos são demasiado desestreturados. É assim necessário uma solução intermédia. De seguida serão analisados algumas formas de suportar estes processos.

Em [16], é apresentada uma metodologia denominada por Unified Activity Management (UAM), que tem como objectivo articular os processos de negócio formais com as colaborações humanas informais necessárias para o suporte de actividades de negócio.

Tipicamente, todo o trabalho que envolva conhecimento é colaborativo, informal e adapta-se a cada situação em particular. Neste contexto, as pessoas usam tecnologias flexíveis de suporte, tais como ferramentas de edição de documentos, email, fórum, e repositórios partilhados que não estão bem integrados com os processos formais. Este tipo de trabalho é referenciado em [16] como actividade de negócio.

As empresas precisam de desenvolver boas práticas para eficazmente reutilizar conteúdos e processos a partir das suas experiências. Contudo, as suas actividades de negócio requerem suporte que não iniba a natureza informal e localmente adaptativa. Este suporte pode ser alcançado através do UAM.

Para a modelação de actividades de negócio num sistema informático, é necessária uma representação explícita e partilhada do conceito actividade de negócio. Assim, o UAM propõe que uma actividade seja representada como uma checklist e que todas as pessoas envolvidas na actividade possam ver e alterar a sua descrição. Uma descrição de actividade articula actores (pessoas) e os papéis nos quais estão envolvidos, os recursos usados (ferramentas, artefactos, pessoas), os resultados produzidos, os eventos a que a actividade está delimitada e as relações com outras actividades (sub-actividades ou outras actividades). A Figura 2.7 ilustra um protótipo de um sistema informático onde a metodologia UAM está patente. No menu do lado esquerdo, "Activity Checklist", está presente uma checklist na qual é possível ao utilizador escolher que actividade ou sub-actividade pretende analisar. Na janela do lado direito, é possível visualizar-se uma janela com o nome "Activity Details View" onde se encontra a descrição da actividade seleccionada (título, estado, recomendações, eventos, pessoas envolvidas e seus papéis, recursos e emails). As sub-actividades principais da actividade RFP 0518 são provenientes de um padrão de actividade e representam as melhores práticas de execução da actividade, adquiridas ao longo da

tempo. Diferentes pessoas, recursos e eventos estão associados a cada uma das actividades e sub-actividades. As pessoas envolvidas numa actividade podem alterar e adicionar sub-actividades, consoante as necessidades da equipa responsável, para responder a um caso em específico.

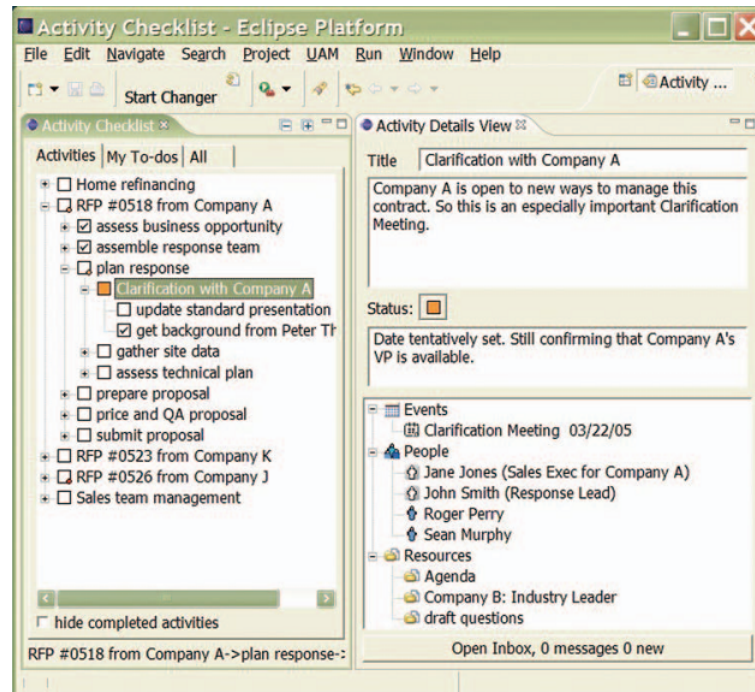


Figura 2.7: Exemplo de uma checklist partilhada num protótipo de uma aplicação UAM

Os objectivos principais desta representação partilhada de actividades são:

- Organizar o trabalho em torno de actividades ao invés de ferramentas ou artefactos - A descrição de actividades disponibiliza todas as informações necessárias para as pessoas desenvolverem as suas actividades.
- Guiar, suportar e coordenar o trabalho, mas não o restringir em demasia - As descrições partilhadas de actividades são guias para realizar as acções devidas, mas as pessoas que controlam a actividade é que decidem se e quando deve ser realizada uma sub-actividade e modificam a descrição de uma actividade para a adaptar à situação actual.
- Disponibilizar um único local para as pessoas gerirem todo o seu leque de actividades - Através da consulta de uma actividade deve ser possível aceder a toda a informação associada à mesma (estado, emails, mensagens escritas, etc), integrando diferentes sistemas.
- Permitir capturar, reutilizar e desenvolver as melhores práticas em actividades padrão - Os padrões de actividade podem ser desenvolvidos através da análise de variações de instâncias, criados a partir do padrão.

A filosofia do UAM consiste em confiar nos mecanismos de interacção social para controlar actividades e impor apenas as restrições que forem necessárias.

Na referência [15] é abordada outra técnica designada por State Flow (SF), que tem como objectivo resolver o problema de flexibilidade existente nos WFMS, considerando para isso uma visão alternativa dos processos de negócio.

A técnica SF tem as suas raízes na interpretação do conceito de processo de negócio que consta em [17]: "um processo de negócio pode ser visto como um conjunto de actividades parcialmente ordenadas e destinadas a atingir um objectivo"(tradução do autor). Esta definição baseia-se na noção de objectivo, o que presume que a qualquer momento da vida do processo é possível aferir se o objectivo do processo foi alcançado ou não. Se não tiver sido atingido, é possível determinar se está longe do alcance. Isto consiste, precisamente, na noção de estado do processo. O estado do processo pode ser final, se o objectivo já tiver sido alcançado, ou intermédio, caso o objectivo ainda não tenha sido atingido. Assim, a noção de estado de processo permite interpretar um processo de negócio como um sistema dinâmico que se move no espaço de todos os estados possíveis, até atingir o estado final (objectivo). Nos processos de negócio os movimentos em direcção ao estado final são realizados através da execução de actividades. No entanto, existem outras fontes de realização desses movimentos, que nem sempre são previsíveis e que podem mover o processo em diferentes sentidos no espaço de estados, p. ex. o cliente alterar a encomenda realizada. A presença de influências externas impossibilita a definição do objectivo do processo como apenas um ponto no estado do processo. O objectivo deve ser definido como um conjunto de estados finais, em que existe um critério que permita concluir que estados finais são atingidos e quais os estados finais que podem ser considerados como mais próximos de um determinado estado intermédio.

O estado actual de um processo, tipicamente, contém informação suficiente para se imaginar que actividades carecem de execução, de forma a que o processo se aproxime do seu estado final. A execução dessas actividades altera o estado do processo e aproxima-o do seu estado final. É óbvio que a execução das actividades necessárias para se atingir um objectivo não pode ser efectuada aleatoriamente. São, assim, necessárias restrições para controlar a ordem pela qual as actividades são executadas.

Em geral, as actividades são planeadas e posteriormente executadas. O planeamento das actividades pode ser usado como uma ferramenta de controlo de execução do processo, ao permitir a criação de uma lista que contém as actividades que devem e são permitidas realizar no estado actual. Após a conclusão de uma actividade do plano do processo, novas actividades podem ser planeadas com base no novo estado do processo. Assim, as restrições sobre a ordem de execução das actividades podem ser definidas como regras do planeamento dinâmico. Para formular essas regras, considera-se o plano do processo como sendo uma parte integral do estado do processo. Isto permite definir a noção de estado válido: estado que inclui todas as actividades requeridas e permitidas para mover o processo para o próximo estado estipulado. Naturalmente que todos os estados finais que possuam um plano vazio pertencem a um estado válido, enquanto todos os estados intermédios com um plano vazio e todos os estados finais com um plano não vazio, não pertencem aos estados válidos.

Com base na noção de estado válido, a ordem de execução das actividades pode ser definida

(p.ex: por um sistema informático) adicionando e removendo actividades no plano, até que um estado inválido se torne válido.

Dependendo do tipo de regras que são introduzidas para controlar o processo, é possível este se encontrar numa situação de controlo pré-definido ou, pelo contrário, num caos completo. Para manter o processo em controlo e, em simultâneo, permitir flexibilidade é necessário proceder a uma estruturação das regras de planeamento. Essa estruturação pode ser efectuada através do uso de políticas. As políticas são por norma divididas em três grupos: obrigações, proibições e recomendações. Exemplos de políticas são:

- Nenhum produto pode ser vendido sem serem efectuados, previamente, testes para aferir o seu bom funcionamento;
- O produto só pode ser levantado pelo cliente, após ter sido efectuado o pagamento;
- Após o cliente pagar o produto, deve-lhe ser explicado em detalhe o funcionamento do mesmo

A classificação das regras em três grupos implica projectar o plano após a execução de uma actividade. Inicialmente, o estado é projectado automaticamente, através da correcção do estado (alteração do plano) em função das regras. No entanto, se o plano resultante for alterado por uma pessoa com privilégios para o fazer, não será possível quebrar qualquer tipo de obrigação ou proibição.

A definição das regras deverá ser efectuada gradualmente. Assim, inicialmente, deve-se definir o processo sem regras, confiando totalmente no planeamento manual. Depois, numa fase mais avançada, quando a natureza do processo estiver mais clara, devem ser adicionadas recomendações. Numa última etapa, algumas recomendações devem ser promovidas a obrigações e algumas proibições adicionadas.

Em [1, 18, 19] está presente outra metodologia designada por DECLARE, que surgiu em oposição às abordagens prescritivas tradicionais de modelação de processos. O DECLARE usa uma abordagem declarativa baseada em restrições, com pontos em comum às regras do SF. Um modelo prescritivo foca-se em especificar exactamente como executar um processo, isto é, todas as possibilidades de execução têm de fazer parte do modelo através da especificação do seu fluxo de controlo. Um modelo declarativo especifica um conjunto de restrições, ou seja, regras que devem ser seguidas durante a execução, definindo assim implicitamente o controlo de fluxo que contém todas as possibilidades que não podem ser violadas.

O DECLARE permite definir os modelos dos processos recorrendo ao DECLARE Designer e dá suporte à execução através do DECLARE Worklist.

A Figura 2.8 ilustra o modelo de um processo de um hotel, efectuado no DECLARE Designer. O modelo consiste num conjunto de actividades e restrições entre actividades. As ligações entre actividades representam restrições provenientes das regras do processo que serão impostas aos utilizadores durante a execução. As restrições podem ser obrigatórias, se forcingem os utilizadores

a executá-las, ou opcionais, se for dada liberdade para o utilizador decidir se as quer seguir ou violá-las.

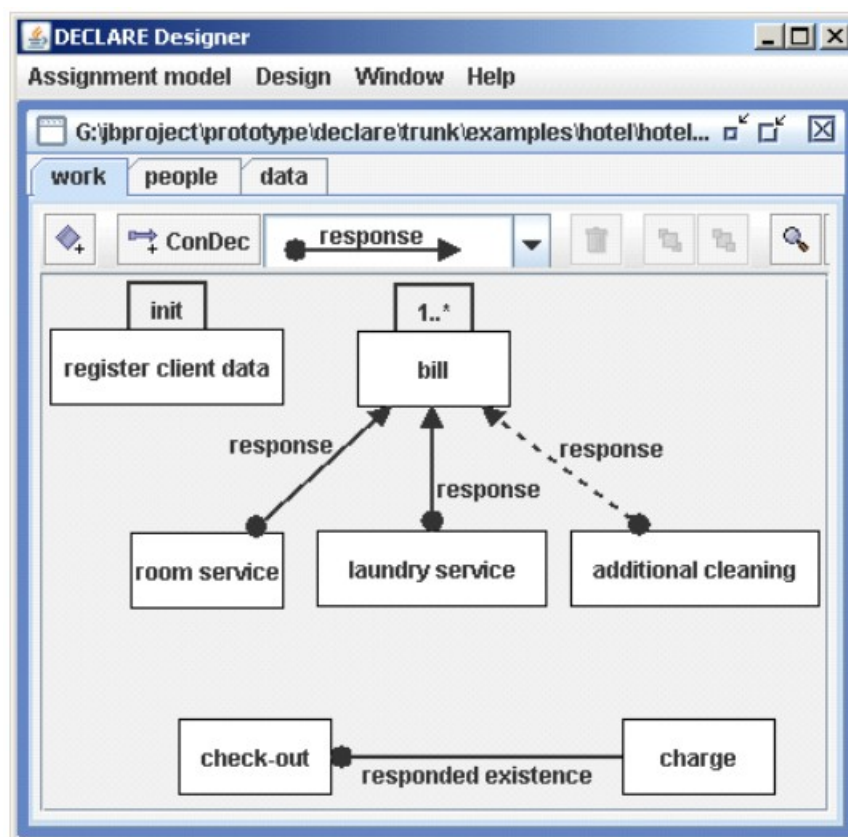


Figura 2.8: Criação de modelo através do DECLARE Designer (fonte: [1])

Após o modelo de um processo ser carregado, os utilizadores podem executá-lo no DECLARE Worklist. A Figura 2.9 mostra a execução do processo do hotel. Uma lista com as instâncias existentes encontra-se do lado esquerdo (*assignments*). O modelo da instância seleccionada é mostrado do lado direito. Podem ser associados dados às actividades caso sejam abertas. Embora a estrutura da instância do processo seja semelhante à do modelo, o Worklist usa símbolos adicionais e cores para ajudar os utilizadores a perceber o estado actual do processo, das actividades e das restrições.

2.2 Sistemas de trabalho semi-estruturados

Nas secções anteriores foram abordados isoladamente os diferentes tipos de processos e os sistemas que os suportam. Porém, o problema é mais complexo do que isso.

Em [20], Steve Alter, define um sistema de trabalho como um sistema em que os participantes humanos e/ou máquinas realizam trabalho usando informação, tecnologia e outros recursos, com o objectivo de produzir produtos e/ou serviços para clientes internos ou externos. Um sistema de

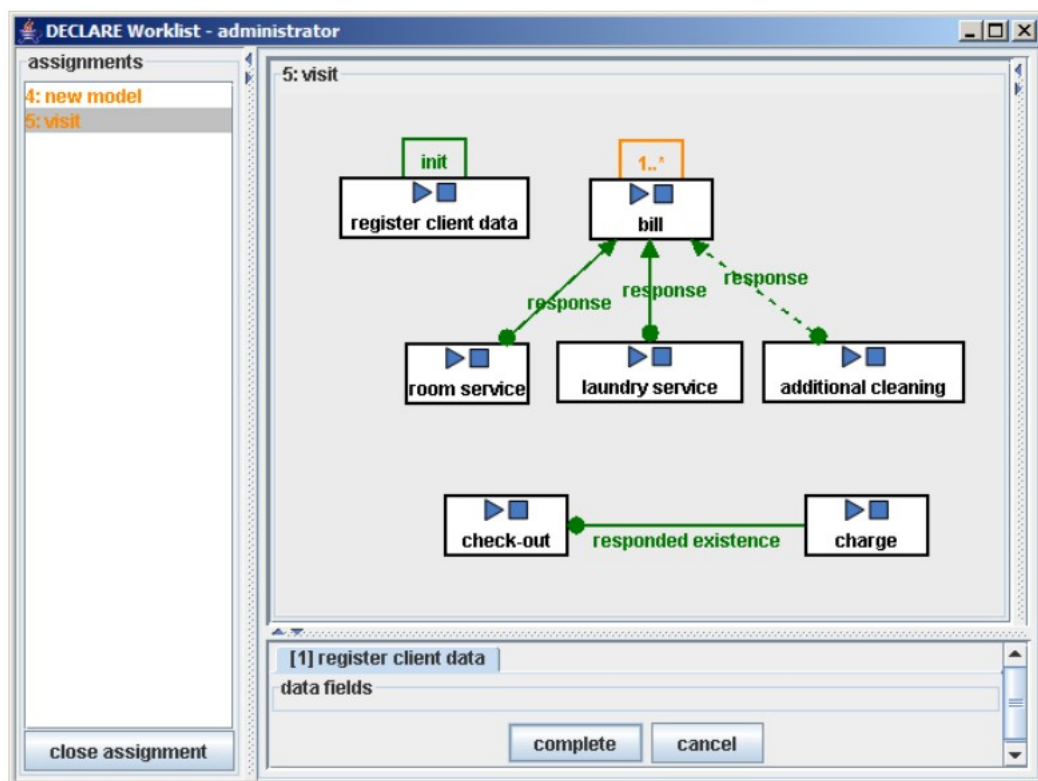


Figura 2.9: Execução de processo através do DECLARE Worklist (fonte: [1])

trabalho pode conter vários processos, cada um deles mais ou menos estruturados (Figura 2.10), o que exige cuidado na concepção do sistema de suporte para o sistema de trabalho em questão.

Quando se pensa em informatizar uma organização, o objectivo principal não é apenas informatizar um processo em questão ou separadamente. Como o sistema de trabalho possui vários processos o objectivo é o sistema informático suportar de uma forma integrada um conjunto de processos com diferentes níveis de estruturação. Para tal, o sistema deve ter capacidade para realizar o seguinte:

- Suportar de forma integrada cada um dos vários processos com diferentes níveis de estruturação: suportar processos de *workflow*, semi-estruturados e colaborativos.
- Permitir gerar uma visão de conjunto (Figura 2.10) dos vários processos: na gestão de uma organização é fundamental existir uma visão de conjunto dos processos e dos seus conteúdos, que permita tirar ilações que digam respeito ao sistema de trabalho como um todo e não apenas a um processo em particular.

Se, por exemplo, uma organização estiver a gerir um processo muito estruturados num sistema de *workflow*, outro processo menos estruturado num sistema como o UAM e outro processo ainda menos estruturado no GoogleGroups, é completamente inviável gerir todos estes processos eficientemente, sendo inclusivamente impossível ter uma visão de conjunto dos vários processos



Figura 2.10: Constituição de um sistema de trabalho semi-estruturado e sua visão de conjunto

da organização, isto porque eles se encontram espalhados por ferramentas distintas para que possam ser suportados convenientemente e em função das suas características.

Quando uma organização possui um sistema de trabalho em que grande parte dos seus processos estão estruturados com um elevado nível de especificação (sistema de trabalho estruturado), então o sistema de suporte a esse sistema de trabalho poderá ser um ERP ou um WFMS, ficando desta forma tudo padronizado. Mas, acontece que há organizações em que os seus sistemas de trabalho são constituídos maioritariamente por processos semi-estruturados (sistema de trabalho semi-estruturado), o que torna a abordagem anterior inadequada nestas circunstâncias.

2.3 Gestão de Conteúdos

Embora a gestão de conteúdos seja um aspecto fundamental em qualquer sistema de gestão de processos e em qualquer sistema de trabalho, nesta secção será dada mais ênfase à gestão de conteúdos para sistemas de trabalho semi-estruturados.

Nos WFMS é comum os conteúdos ficarem confinados ao processo que os deu origem, como ilustra a Figura 2.11. Mas nem sempre esta abordagem é a ideal.

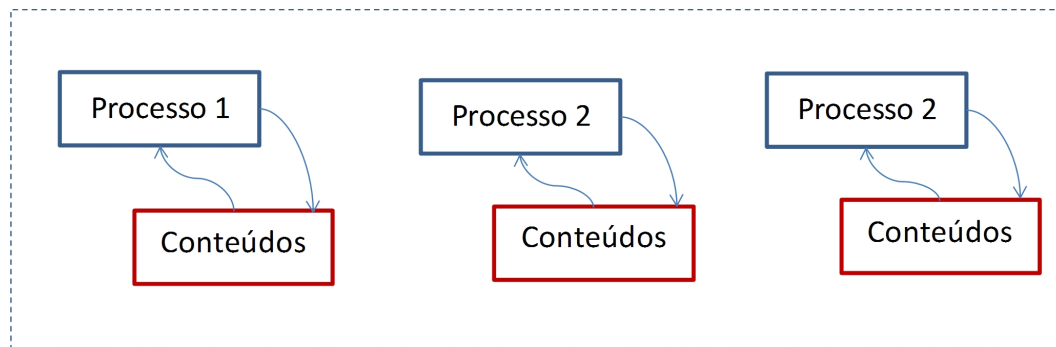


Figura 2.11: Gestão de conteúdos num WFMS

Os processos existentes em maior quantidade num sistema de trabalho semi-estruturado (processos semi-estruturados) são tipicamente baseados no conhecimento, isto é, na experiência e *know-how* dos actores envolvidos. Isso faz com que esteja por norma associado a estes processos muita informação e conhecimento, sendo assim processos que têm como característica intrínseca o facto de possuírem muitos conteúdos associados, e.g., consultório de advogados ou empresa de engenharia. Como os processos são semi-estruturados e não têm grande rigidez, os conteúdos podem ser ficheiros em diferentes formatos (pdfs, documentos Word, documentos Excel) ou emails, ficando este tipo de opções a cargo das preferências dos actores intervenientes. Assim, a informação pode ser proveniente de diferentes fontes e de diferentes suportes, e.g., uma decisão tomada tanto pode estar em pdf como em email. É então necessário um sistema de gestão de conteúdos que consiga no âmbito de um processo agregar e integrar conteúdos de diferentes origens e diferentes suportes.

Contudo, neste contexto, é comum pretender-se aceder não apenas a conteúdos que fazem parte directamente do processo em questão mas sim de outros processos ou entidades, isto porque os documentos têm valor por si. Por outras palavras, quando se está a gerir uma instância de um processo de contratação poderá ser necessário aceder, por exemplo, aos contratos realizados no âmbito de outros processos. Isto realça o facto de que quando for finalizado um processo que deu a origem a um contrato e quando for necessário posteriormente aceder a esse contrato, o mais importante já não é o processo que deu origem ao documento mas sim o próprio documento e o conhecimento que ele contém.

A Figura 2.12 ilustra como podem ser geridos os conteúdos num sistema de trabalho semi-estruturado, de forma a permitir o acesso a conteúdos que não correspondem directamente a um

dados processos. Os processos podem ter associados um conjunto de conteúdos, tal como acontece com os processos 1, 2 e 3. Se esses conteúdos estiverem associados a tópicos (certificados, contratos, etc...) futuramente é mais fácil de reutilizar o conhecimento. É isto que faz o processo 3, que recorre aos conteúdos que se encontram no tópico 2. Com esta abordagem é possível aceder a conhecimento sem ser necessário para isso aceder ao processo que o gerou.

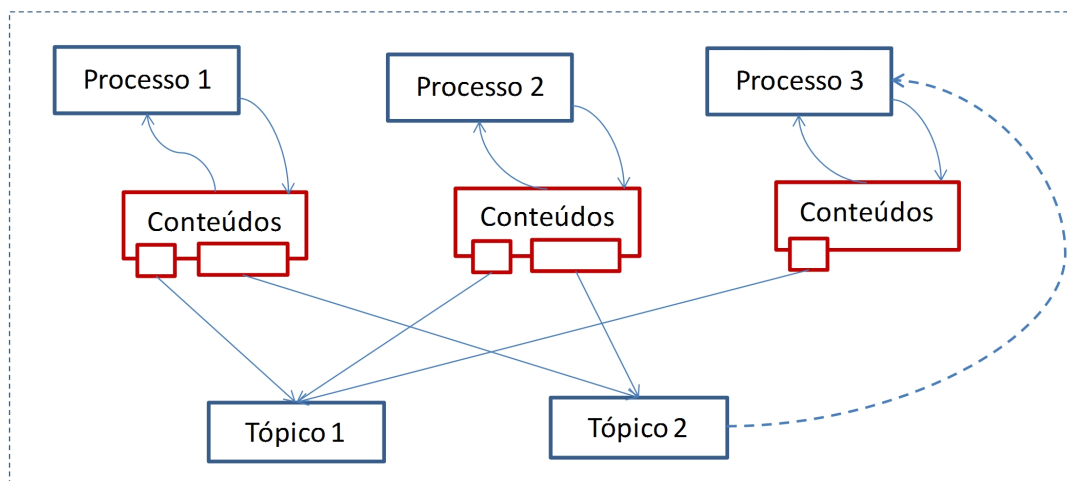


Figura 2.12: Gestão de conteúdos num sistema de trabalho semi-estruturado

2.4 Síntese

As organizações actualmente para fazer face a diversos problemas recorrem ao uso de processos. Posteriormente, vêem nos sistemas informáticos o caminho mais natural para aumentar a eficiência.

A implementação de um sistema informático de suporte a uma organização não é uma tarefa trivial e depende muito das características do trabalho desenvolvido. "Fazer com que as pessoas usem um novo sistema correctamente é muito mais difícil do que desenvolver o sistema e colocá-lo em operação"(Steve Alter, 2006). De facto o difícil não é desenvolver um sistema de informação, difícil é desenvolver um sistema que se adapte aquilo que é feito numa organização e que facilite o trabalho, de forma a aumentar a produtividade. Para atingirmos esse objectivo, é fundamental ter presente que existem processos com diferentes níveis de estruturação e consequentemente com diferentes sistemas de suporte.

Porém, o verdadeiro desafio das organizações não passa por informatizar um único processo, mas por conseguir um sistema que permita, de uma forma integrada, a informatização de processos com diferentes níveis de estruturação. Para além de gerir os processos, é fundamental o sistema gerir os conteúdos associados e ser flexível na forma como permite o acesso aos conteúdos, sendo que se torna também um desafio a gestão de documentos quando estão em causa processos com diferentes graus de estruturação.

Capítulo 3

Análise do sistema de trabalho

Neste capítulo pretende-se estudar e extrair as características fundamentais do sistema de trabalho semi-estruturado, com o objectivo de encontrar um conjunto de requisitos que um sistema de suporte a este sistema de trabalho deve satisfazer. Para tal, utilizar-se-á como caso de estudo os STM da FEUP, que como veremos, é representativo de um elevado número de sistemas com estas características.

O capítulo começa por descrever, na Secção 3.1, os STM ao nível da sua organização, processos e actividades, de modo a que sejam perceptíveis as características inerentes a um sistema de trabalho semi-estruturado. Para tornar ainda mais elucidativo, na secção seguinte, são analisados dois processos semi-estruturados pertencentes a este sistema.

De seguida, na Secção 3.3, serão descritas as necessidades que um sistema de suporte semi-estruturado deve satisfazer, com base na informação contida nas duas secções anteriores. Também são identificados os módulos necessários para suportar este sistema de trabalho, tendo em conta o actual sistema de suporte dos STM, que será o alvo de uma futura implementação.

3.1 Organização e actividades

Os sistemas de trabalho semi-estruturados são tipicamente encontrados nas empresas de serviços baseados no conhecimento, de que os gabinetes de engenharia são um exemplo.

Os Serviços Técnicos e de Manutenção (STM) da FEUP oferecem também um bom exemplo de sistema de trabalho semi-estruturado. A sua actividade é exercida numa área total de 84 400 m^2 constituída por 14 edifícios diferentes (ver Figura 3.1) e distribui-se por três áreas de actividade:

- **Gestão da manutenção:** manutenção "clássica" de equipamentos e sistemas (aquecimento, ar condicionado, bombagem, PT's, elevadores, etc.);
- **Gestão de projectos e empreitadas:** novo sistema CCTV, novo sistema controlo de acessos, obras de pequena, média e "grande" dimensão, etc.;

- **Gestão das operações correntes:** limpeza, vigilância, apoio às salas de aula, reserva de salas, gestão de resíduos, etc.



Figura 3.1: Área de actuação dos STM

Os STM encontram-se divididos por unidades, para que cada uma realize actividades com afinidades entre si. Na Figura 3.2 está presente a estrutura organizacional dos STM. É possível constatar que os STM são compostos por quatro grandes unidades. Contudo, a unidade "Equipamentos, sistemas e infra-estruturas" está dividida em duas (ESI e IEE) por conter equipamentos que podem ser eléctricos ou electromecânicos. O ESI é responsável pelos equipamentos eléctricos e, por sua vez, o IEE pelos electromecânicos. Logo, as unidades que compõem os STM são: EDE, ESI, IEE, HSA e SGR. Cada uma possui um diferente gestor e consequentemente áreas de responsabilidade distintas.

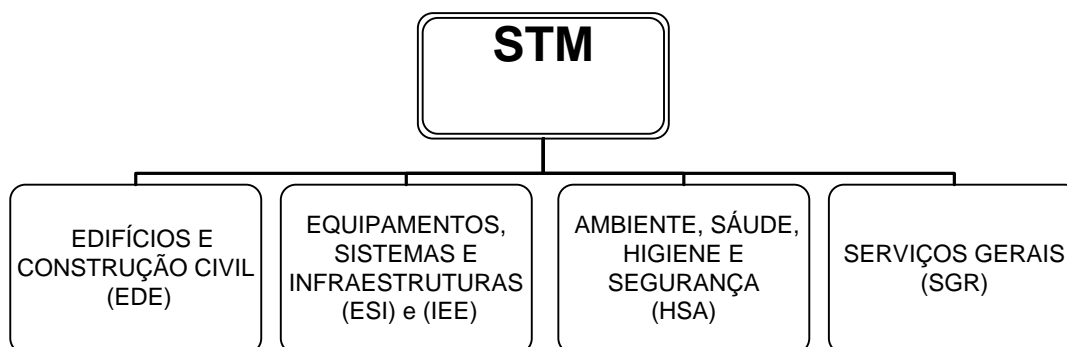


Figura 3.2: Estrutura organizacional dos STM

A Tabela 3.1 contém o âmbito de actuação de cada unidade. Como é possível constatar, as características das actividades variam muito de unidade para unidade. Isto faz com que a gestão de cada unidade também varie com base nos seguintes factores:

- a natureza dos equipamentos;
- características dos processos (elaborados apenas por recursos internos, apenas por recursos externos, ou mistos);
- requisitos legais a satisfazer;
- informalidade na gestão da unidade.

Como dá para constatar através do Anexo A, nos STM existe uma elevada quantidade de equipamentos. Para além disto, tratam-se de equipamentos de naturezas distintas. Tais factos, incutem uma elevada variabilidade nas actividades desempenhadas em cada unidade e consequentemente uma elevada complexidade na gestão dos seus processos. Não é possível definir-se um processo que dê para todos os equipamentos, mas também nem sempre é possível ter um processo que cubra, p.ex., todas as manutenções de um equipamento em questão, porque embora o equipamento seja o mesmo, as tarefas a executar em cada manutenção podem variar bastante. Estamos então perante dois tipos de variabilidade nos processos: ao nível das instâncias de um processo e ao nível dos processos entre equipamentos, por o processo variar muito para um equipamento ou entre um conjunto de equipamentos, respectivamente. Estes processos são semi-estruturados, porque é difícil definir-se um modelo rígido, uma vez que estão sujeitos a grandes variabilidades.

Quando se considera, por exemplo, a manutenção das UPS, do ar comprimido e dos chillers, estamos perante um conjunto de manutenções que variam devido à natureza dos equipamentos. Portanto, poderia ser definido um processo para cada equipamento e que fosse adaptável a cada instância, ou um processo "genérico" que fosse adaptável a diferentes equipamentos e, posteriormente, a cada instância. Se os processos variarem bastante de equipamento para equipamento, talvez o melhor seja definir-se um modelo para cada equipamento e fazer-se uma adaptação apenas ao nível da instância do processo, se tiverem alguns pontos em comum, talvez se possa definir um modelo único que facilmente se adapte a cada equipamento e posteriormente à instância. Mas acontece que nos STM existe uma elevada quantidade de equipamentos o que inviabiliza a definição de um modelo de processo para cada equipamento, porque iríamos ter várias dezenas de modelos. Posto isto, a melhor abordagem passa por ter um modelo processo ou um número limitado de modelos de processos que sejam suficientemente flexíveis para adaptar a diferentes equipamentos e posteriormente à instância.

Os equipamentos que fazem parte dos STM (ver Anexo A) podem estar sujeitos a três tipos de manutenção:

- manutenção com base em recursos internos (MRI);
- manutenção curativa com base em recursos externos (MCRE);
- manutenção programada com base em recursos externos (MPRE).

EDIFÍCIOS E CONSTRUÇÃO CIVIL (EDE)
Edifícios Outras estruturas (poços, depósitos água, galeria técnica, redes enterradas passivas) Parques de estacionamento Áreas verdes
EQUIPAMENTOS, SISTEMAS E INFRA-ESTRUTURAS (ESI e IEE)
Redes normais: <ul style="list-style-type: none"> • eléctrica • abastecimento de água potável e não potável • gás combustível • informática/telecomunicações Redes especiais: ar comprimido, gases de laboratório Sistemas AVAC (aquecimento, ventilação e ar condicionado) Sistemas de emergência (grupos electrogeradores e UPS) Elevadores Sistema de Gestão Técnica Centralizada: <ul style="list-style-type: none"> • Supervisão e controlo de equipamentos • Detecção de incêndios • Controlo de acessos
AMBIENTE, SAÚDE, HIGIENE E SEGURANÇA (HSA)
Encaminhamento de resíduos domésticos e laboratoriais Limpeza corrente e casos especiais Segurança (safety) Higiene e ergonomia Saúde no trabalho Tratamento/desinfecção de água Desinfestações
SERVIÇOS GERAIS (SGR)
Vigilância e segurança (security) Apoio às salas de aula: <ul style="list-style-type: none"> • Preparação dos equipamentos audiovisuais • Fornecimento de consumíveis • Manutenção do mobiliário • Configuração das salas para usos específicos Fornecimento de consumíveis para sanitários Movimentação de equipamentos e outros bens e de resíduos

Tabela 3.1: Âmbito de actuação das unidades pertencentes aos STM

Grande parte dos equipamentos estão sujeitos a manutenção (curativa e/ou programada) com base em recursos externos. Daí que a "Gestão da subcontratação" (Tabela 3.2) seja uma das actividades internas com maior relevância. De facto, gerir uma equipa externa subcontratada é completamente diferente de gerir uma equipa interna e tem impacto no sistema de trabalho e no sistema de suporte.

GESTÃO DE PROJECTOS
Elaboração de cadernos de encargos/termos de referência Processo de escolha do fornecedor (concurso, ajuste directo) Formalização do contrato Gestão da obra (controlo administrativo, acompanhamento, resolução de problemas técnicos)
GESTÃO DA SUBCONTRATAÇÃO
Elaboração de termos de referência Processo de escolha do fornecedor (concurso, ajuste directo) Formalização do contrato Acompanhamento e resolução dos problemas
GESTÃO DE ACTIVIDADES CORRENTES
Gestão diária dos recursos humanos Avaliação de necessidades e acompanhamento das pequenas intervenções (redes eléctrica e estruturada, sistemas AVAC, elevadores, manutenção dos edifícios, etc.) Apoio às salas de aula (actividade dos contínuos) Acompanhamento dos serviços de limpeza e vigilância Gestão e controlo das comunicações telefónicas e de dados (redes fixa e móvel: PT, Vodafone, Optimus, TMN) Gestão das "utilities" (gás, água, electricidade, comunicações)

Tabela 3.2: Principais actividades internas dos STM

Os STM dispõe actualmente de 24 contratos com empresas externas, conforme descrito na Tabela 3.3, todos eles revistos anualmente, com plano de manutenção, com controlo das operações executadas, etc. Para além de serem bastantes contratos, cada um tem as suas características e pertence a equipamentos de diferentes naturezas.

Grande parte dos processos a gerir dizem respeito a trabalhos efectuados por entidades externas (sub-contratadas). Logo, não podem ser geridos da mesma forma que seriam caso fossem trabalhos desenvolvidos por equipas internas, em que era possível ter-se um sistema

EQUIPAMENTO/SISTEMA	EMPRESA
Barreiras parques	Soltráfego
Elevadores	Schindler
Compressores	Atlas Copco
Centrais Térmicas	Anemos-Malbar Magalhães
Roof top's	Lennox
Chillers da Biblioteca	Nónio Hiross
Ar condicionado - CICA	Nónio Hiross
Torres arrefecimento Biblioteca	Tempo Invertido
AVAC(Limp. Filtros)	Montequi
Termoacumuladores	Feio & Comp. SA
Insp.Rede de gás	ISQ
Bombas de água	Grundfos
Vigilância	Viprese
Limpeza	Limpotécnica
Desinfestações	Rentokil
Ambientadores sanitários	Initial
Contentores assépticos	Initial
Postos de transformação	Efacec
Central telefónica	PT Prime
Gestão Técnica Centralizada	Microprocessador
Grupos geradores	STET
UPS	MGE
Detectores de hidrogénio	ArLíquido
Sistemas de segurança (SADI, LMS, CCTV, controlo de acessos)	Siemens

Tabela 3.3: Empresas responsáveis pela manutenção dos equipamentos dos STM

tradicional padronizado e todos os colaboradores sabiam como interagir com ele e faziam-no todos da mesma forma. Como estes processos possuem actores externos à organização e eles não podem interagir com o sistema de informação directamente, essa interacção tem de ser efectuada com base na troca de emails entre actores internos e externos. Contudo, por vezes não basta a troca de emails, é necessário enviar documentos anexados com informação adicional, e.g., planos de manutenção. Mas como cada fornecedor tem o seu próprio plano de manutenção, é muito difícil integrar e uniformizar a informação proveniente dos mesmos. Tem-se assim de gerir N emails e N ficheiros associados aos processos, sendo uma informação por natureza muito menos estruturada comparativamente à informação que poderia ter um sistema interno em que toda a gente o usava. É então comum os processos terem informação que pode estar numa base de dados, informação em emails e/ou informação em ficheiros, podendo a informação estar espalhada por vários suportes no âmbito de um processo. Por exemplo, para um processo, pode-se querer saber se foi trocado um email com o fornecedor a combinar uma reunião (conteúdo em email), consultar o contrato realizado (conteúdo em ficheiro), inserir um registo na agenda para ser lançado um

alerta posteriormente no plano de manutenção (conteúdo em base de dados), etc. Quando estamos a gerir um processo queremos ver todos os conteúdos independentemente dos seus suportes.

Porém, como os conteúdos têm valor por si, frequentemente é necessário consultá-los em diferentes contextos para além dos processos, e.g., quando se está a controlar os pagamentos a um fornecedor pretende-se saber o que já foi pago independentemente dos processos. O mesmo documento pode ser alvo de consulta no âmbito de um processo ou noutros contextos.

Existe alguma informalidade na forma como são geridas as unidades, que varia de gestor para gestor. Alguns gestores preferem gerir a suas áreas de responsabilidade com menor estruturação, outros com maior. Um gestor pode querer organizar os seus documentos por dia, outro por ano e outro por tipo de equipamento. Deve-se dar a capacidade para o sistema ser flexível para se adaptar às características dos próprios equipamentos, mas também às preferências do utilizador. Isto faz com que haja uma variabilidade associada aos equipamentos, mas também associada à forma como são geridas as unidades. Contudo, a estruturação dentro de cada área deve ser flexível, mas dentro de um limite que permita gerar uma visão consolidada do que se está a passar nos serviços, ou seja, se a unidade ESI for gerida de uma maneira e a IEE de outra diferente, internamente a gestão pode ser diferente e com diferentes graus de estruturação, mas isso não deve impedir que haja uma visão consolidada que permita ao gestor do serviço saber o estado de cada unidade, gerando uma visão de conjunto. Estes problemas não são apenas específicos dos STM, em muitos processos e em muitas organizações onde o conhecimento é relevante, e.g., gabinete de engenharia ou de advogados, encontram-se este tipo de situações. Cada advogada gere os seus processos à sua maneira, mas isso não implica que não possa haver um controlo global que permita saber o estado dos processos ao nível central. Para isso, tem-se de ser capaz de agregar informação proveniente de diferentes locais e com diferentes estruturas e apresentá-la numa interface consolidada para quem está a gerir. Isto só é possível se existirem alguns pontos em comum entre os processos, daí a importância da semi-estruturação: deve-se estruturar até certo ponto para permitir consolidar a informação, mas é necessária em simultâneo uma flexibilidade que permita às pessoas adaptarem o seu trabalho às características de cada processo.

3.2 Análise de processos

Nesta secção, são analisados dois processos fundamentais dos STM: o processo *Executar intervenção de manutenção* da área de gestão da manutenção e o processo *Realizar projecto/empreitada* da área da gestão de projectos, com o objectivo de compreender melhor as características dos processos semi-estruturados e permitir identificar os requisitos para os sistemas de suporte à sua execução.

3.2.1 Processo Executar intervenção de manutenção

Embora os processos associados às intervenções de manutenção sejam relativamente simples, existe uma grande variabilidade no seu fluxo de execução, o qual depende de múltiplos factores como o tipo de equipamento, o tipo de intervenção e o tipo de avaria. Assim, por exemplo:

- A execução da intervenção pode ter sido prevista previamente ou ser despoletada após a ocorrência de uma avaria;
- A intervenção pode requerer um planeamento prévio para acerto de data e de recursos com a entidade prestadora do serviço;
- A intervenção pode envolver tarefas de preparação, por exemplo, aviso aos utilizadores no caso do sistema ser retirado de serviço durante a intervenção, isolamento e protecção da zona que será objecto da intervenção, activação de meios de substituição;
- A intervenção pode ser efectuada ao abrigo de um contrato de manutenção, ou ser objecto de uma aquisição de serviços específica;
- A intervenção pode envolver custos adicionais e, consequentemente, implicar o processamento de uma despesa, conforme a necessidade de se proceder à substituição de peças;
- Conforme o montante envolvido, a despesa pode estar sujeita à apresentação de um orçamento por parte da entidade externa e a sua aprovação pelos STM ou, em casos especiais, pelo próprio Director da FEUP;
- Conforme a urgência da intervenção, alguns dos passos na execução de uma intervenção, podem ser "curto-circuitados". É tipicamente os casos dos procedimentos de aprovação e autorização de despesas que, em situações de urgência, podem ser tratados de forma mais expedita.

A Figura 3.3 mostra o fluxo de execução típica do processo Executar intervenção de manutenção. Conforme a descrição anterior, a execução de cada instância do processo pode desviar-se deste fluxo, seja porque algumas tarefas não são necessárias, e.g., preparação prévia, seja porque a urgência do processo leva a não as considerar, e.g., aprovação formal do orçamento.

Tal como acontece com as tarefas, também os documentos e informações associados a cada intervenção particular podem diferir. Exemplos típicos dos documentos que podem constar no dossier de uma intervenção são os seguintes:

- Folhas de obra com descrição dos trabalhos efectuados
- Facturas
- Orçamentos
- Mensagens de email trocadas entre o prestador de serviço e os STM
- Documentação técnica dos componentes e equipamentos objecto da intervenção

Para cada um destes documentos, pode justificar-se conservar mais do que uma versão. Por exemplo, no caso dos orçamentos, é possível que uma primeira proposta do prestador de serviço seja objecto de negociação e que dê origem a uma segunda versão do orçamento. Em alguns casos,

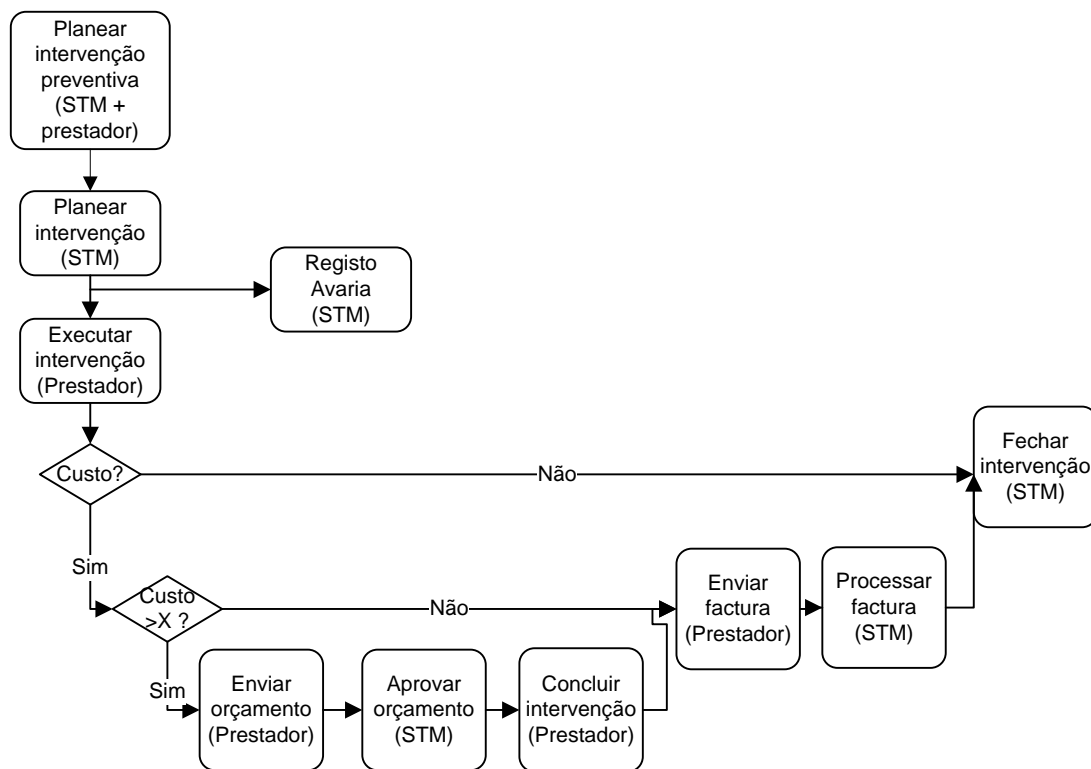


Figura 3.3: Processo Executar intervenção de manutenção

também pode acontecer que a alteração ao orçamento inicial não dê origem a um novo orçamento mas que certas condições adicionais sejam acordadas directamente entre as pessoas envolvidas e fiquem registadas, e.g., através do arquivo de mensagens de email.

Apesar destas possibilidades, o processo Executar intervenção de manutenção não deixa de ser um processo de trabalho simples e relativamente ao qual seria possível proceder a um análise exhaustiva e prever todas as situações possíveis que podem ocorrer na execução de uma dada instância do processo. No entanto, se se atender à diversidade de equipamentos e infraestruturas que podem ser objecto das intervenções, à gravidade das avarias e à urgência das reparações, aos vários intervenientes internos e externos que podem estar envolvidos, se se tentasse representar tudo no mesmo modelo, o modelo do processo que resultaria desse exercício seria, provavelmente, demasiado complexo para ter interesse prático.

Como tal, a ideia de modelar completamente um processo por forma a ser possível controlar a sua execução através de sistemas de gestão de workflow tradicionais, que impõem um fluxo rígido de execução, pode revelar-se contraproducente.

Uma abordagem que pode ser mais eficaz consiste em oferecer uma ferramenta de execução mais flexível que, em vez de impor uma trajectória rígida na execução do processo, apenas imponha como limites à execução do processo um conjunto de restrições relativas àquilo que terá obrigatoriamente de ser respeitado. Por exemplo, não pode ser efectuado nenhum pagamento sem se estar na posse da respectiva factura e sem que esta tenha sido validada.

Nesta abordagem o responsável pela execução do processo conserva um significativo grau de liberdade para que, em cada situação, possa optar pela solução que lhe parecer mais adequada, solução essa que poderá passar, em caso de dúvida, pela transferência da responsabilidade pelo encaminhamento do processo para uma instância de decisão de nível superior.

Nesta perspectiva, o sistema de gestão não impõe um fluxo de execução mas, em vez disso, limita-se a sugerir a trajectória que, desejavelmente e em condições normais, deverá ser seguida na execução do processo, sem restringir desnecessariamente a liberdade do responsável.

Dito de outra forma, em vez de impor uma estruturação rígida do fluxo, o sistema de gestão do processo deve semi-estruturar a execução do processo, verificando o cumprimento das restrições com carácter obrigatório e guiando o responsável dentro dos limites impostos por essas restrições.

3.2.2 Processo Realizar projecto/empreitada

A análise anterior teve por objecto um processo relativamente simples. No caso de um projecto/empreitada o problema é, naturalmente, muito mais complexo. Tipicamente, um projecto/empreitada envolve 3 grandes fases:

- Projecto onde, numa primeira fase, estão envolvidos os arquitectos e, depois, os técnicos das várias especialidades e que termina com a elaboração do caderno de encargos.
- Concurso conduzida pelos STM e que conduz à selecção do prestador de serviço que irá executar a obra.
- Obra onde se procede à realização material da obra. Tipicamente, a obra será executada por um prestador de serviço externo com acompanhamento e fiscalização dos STM, ou de uma entidade externa contratada para o efeito.

Em paralelo com a execução física do projecto/empreitada decorre a execução financeira, a qual contempla a elaboração de orçamentos e o processamento das despesas aos vários prestadores de serviços envolvidos.

Algumas fases do processo estão fortemente regulamentadas, quer por legislação pública quer por normas e procedimentos internos. É o caso da fase Concurso regulamentada pelo Código de Compras Públicas que impõe de forma muito detalhada os procedimentos e os documentos a respeitar na aquisições de bens e serviços e nas empreitadas.

Ao contrário, há outras fases do processo que são muito menos "normalizáveis". Embora possam ser definidas políticas e orientações, o fluxo de execução é largamente determinado por uma avaliação, em cada caso, da situação do processo. São relativamente frequentes as situações em que, devido à alteração da conjuntura em que decorre o projecto (e.g., o aparecimento de uma nova necessidade), pode ser necessário introduzir alterações no projecto, tanto ao nível das especificações como dos prazos de execução.

Essas alterações podem alterar significativamente o fluxo do processo ao nível operacional, e.g., ao nível das tarefas, pois o processo pode ter de regressar a fases anteriores, podem ser necessárias novas tarefas (e.g., de replaneamento ou de renegociação com os fornecedores).

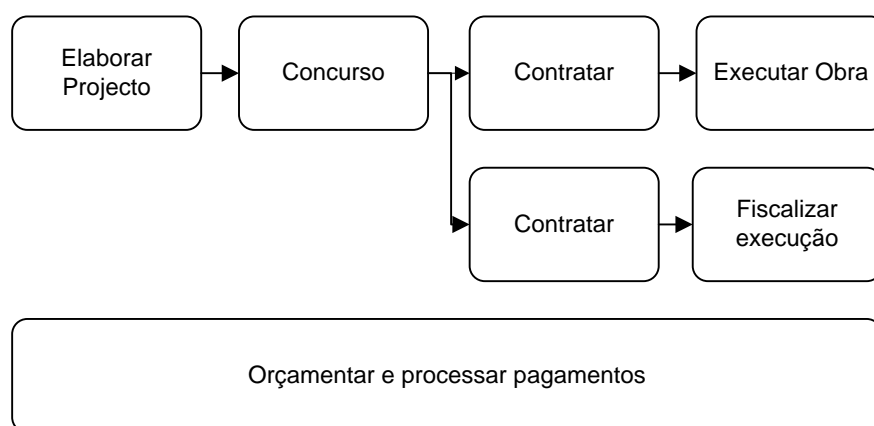


Figura 3.4: Processo Realizar projecto/empreitada

Nestes casos, embora deva sempre ser feito um esforço para normalizar procedimentos, muitas vezes, "as exceções são a regra" na execução dos processos e, embora fosse teoricamente possível proceder a uma análise exhaustiva de todas as opções com que se pode deparar o responsável pela gestão do processo, na prática, isso é irrealizável.

Uma abordagem alternativa será a de definir apenas a sequência normal (ou "desejável") das operações e acompanhá-la de um conjunto de políticas, orientações e regras que, em cada situação, estabeleçam aquilo que é obrigatório, aquilo que é recomendável e aquilo que é possível fazer.

Este princípio aplica-se tanto ao fluxo do processo como à documentação que lhe está associada. Se, num concurso público, certos documentos são obrigatórios por lei (e.g., caderno de encargos, convites aos candidatos à prestação do serviço, actas das reuniões), muitos outros não são obrigatórios (actas, ofícios e anotações internas que justifiquem eventuais opções) mas podem ser fundamentais para uma correcta compreensão do rumo seguido na execução do processo.

A breve análise dos processos Executar intervenção de manutenção e Realizar projecto/empreitada aqui apresentada permite concluir que aquilo que se espera de um sistema de gestão de semi-processos não é a imposição de um fluxo de execução rígido e automatizado, como acontece nos sistemas de gestão de workflow convencionais, mas sim um conjunto de ferramentas e de informações que apoiem o responsável por cada instância do processo a atingir os seus objectivos da forma mais eficiente possível e respeitando as regras e restrições com carácter obrigatório.

Entre essas informações contam-se o modelo do processo e as respectivas instruções de trabalho, a legislação aplicável, os dossiers de outras instâncias idênticas do processo, o plano global do processo e o registo das tarefas criadas durante a sua execução, as mensagens de email trocadas entre os intervenientes do processo, os *deliverables* do próprio processo, anotações e outras informações sobre o processo que tenham sido registadas pelos seus intervenientes.

Um dos princípios chave dos paradigmas de trabalho dos processos semi-estruturados passa por dar ao utilizador alguma liberdade na execução dos processos, afinal a experiência adquirida por ele é muito importante e contribui para a melhoria do trabalho realizado e por conseguinte para o aumento da produtividade.

Durante a execução dos processos acima analisados é possível detectar uma série de obstáculos invisíveis que podem contribuir para a diminuição da produtividade:

- Navegação entre vários sistemas: email, sistema de calendarização, sistema de avisos, editor de texto, facturas, telefone, etc.
- Copiar e colar: Manter o registo das acções tomadas em Excel (actualmente o método utilizado para manter os processos organizados).
- Metadata na informação: arquivar documentos é essencial mas tem que ser feito manualmente. Uma factura pode estar por pagar ou não, o utilizador precisa de analisar o documento e só arquivar depois de decidir o estado da mesma, ou seja, nos sistemas normalmente usados existe falta de semântica na informação trocada entre sistemas. Um humano facilmente percebe o que é um endereço de email, no entanto, a menos que o sistemas esteja desenvolvido nesse sentido não detecta o que leva a perda de tempo acrescidas.
- Funcionário como integrador de tecnologias ligando todos os métodos e tecnologias, o que faz com que despenda demasiado tempo a transportar e procurar informação através das várias ferramentas utilizadas.

Como podemos ver a divisão do conteúdo, estado dos processos e planeamento são problemas reais quando tudo se encontra separado por diversos tipos de aplicações e ferramentas. Os processos têm que evoluir de forma a poderem funcionar num ambiente descentralizado e onde o utilizador tem sempre um papel muito relevante.

A única forma de tirar máximo partido de todos os serviços é apresenta-los de tal forma que os utilizadores facilmente os compreendam e que sejam acessíveis e os achem como uma extensão natural dos seus hábitos normais de trabalho. O objectivo é fazer com que seja fácil aceder aos serviços do contexto em que o trabalho se está a realizar, criando pontes e mecanismos que reduzam o factor de integração do utilizador que até agora vinha a ser um obstáculo.

3.3 Análise das necessidades

Nos STM existe um conjunto de aspectos a ter em conta aquando o desenvolvimento do sistema de suporte:

- Gestão integrada da manutenção e dos projectos (empreitadas)
- Solução flexível
- Adaptação à organização da FEUP

É fundamental a gestão da manutenção e dos projectos estarem integradas num único sistema. Tipicamente a mesma pessoa gere projectos e manutenções, logo não está interessada em ter dois

sistemas distintos para o fazer. O objectivo não pode consistir em ter "apenas" um Sistema de Gestão da Manutenção (SGM), mas antes um sistema que permita a gestão integrada das várias áreas de actividade dos STM:

- Manutenção;
- Projectos/Empreitadas;
- Operações correntes.

Para tal, o sistema deve gerir de forma integrada:

- As ordens de trabalho;
- Toda a documentação (manuais, legislação, cadernos de encargos, propostas, contratos, etc.);
- As aquisições de bens e serviços.

O aspecto "solução flexível" é fundamental porque, como vimos anteriormente, existe uma elevada variabilidade nos processos. Os projectos com que os STM lidam têm uma dimensão e grau de complexidade muito variável, e.g., pequenas intervenções e empreitadas de média/grande dimensão. O sistema deve ser suficientemente flexível para possibilitar a gestão dos processos mais complexos sem se tornar demasiado pesado no caso dos processos mais simples. A solução também tem de ser flexível ao nível da gestão de conteúdos, possibilitando a gestão flexível de cada unidade.

É igualmente importante que o sistema se adapte à organização da FEUP. Não se pode impor um sistema que altere a organização existente. As pessoas estão organizadas de uma determinada maneira e essa maneira deve reflectir-se directamente no sistema, isto é, não devem ser as pessoas e o trabalho a realizar a adaptarem-se ao sistema, mas sim o sistema a adaptar-se às pessoas e às características do trabalho.

A análise da Secção 3.1 e da Secção 3.2, permite extrair um conjunto de características fundamentais inerentes a sistemas de trabalho semi-estruturados:

- **grande diversidade de processos:** processos simples, e.g., manutenção de equipamento, até processos mais complexos, e.g., projectos de empreitadas; a existência de bastantes equipamentos de diferentes naturezas implica muitos processos distintos;
- **grande diversidade na gestão de processos referentes à mesma entidade:** a forma como é gerido o mesmo equipamento varia de manutenção para manutenção;
- **a forma como são geridos os processos depende dos seus actores:** cada actor gere as suas actividades da forma que achar mais pertinente, embora tenha de existir sempre um conjunto de factores em comum para todas as actividades;

- **elevado número de actores externos intervenientes nos processos:** dificulta a uniformização dos dados; obriga a troca de *emails* e ficheiros como forma de interacção, o que implica a existência de conteúdos em diferentes suportes;
- **conteúdos têm valor em diferentes contextos:** tanto pode ser necessário aceder a conteúdos no âmbito de um processo, como noutro contexto diferente.

Assim, um sistema de suporte que dê resposta a estas características e que tenha em conta os aspectos anteriormente mencionados, terá de satisfazer as seguintes necessidades:

- **permitir a gestão de processos com diferentes níveis de estruturação de forma flexível:** para que o processo possa ser adaptado a diferentes equipamentos, às instâncias, às necessidades de cada unidade e aos actores internos, mantendo a capacidade de consolidação de conteúdos;
- **permitir uma visão consolidada das unidades,** para que seja possível o gestor da unidade saber o que se está a passar na sua área de responsabilidade;
- **permitir a gestão integrada de diferentes áreas de actividade,** e.g., projectos e manutenção;
- **integrar na mesma interface informação proveniente de diferentes suportes** (*emails*, ficheiros, base de dados, etc.);
- **permitir consultar conteúdos em diferentes contextos,** e.g. consultar o mesmo certificado dentro do processo que o deu origem ou noutros processos.

Nos STM tem-se vindo a desenvolver uma aplicação integrada de suporte às suas actividades, baseada em SharePoint. No momento em que teve início esta dissertação o sistema oferecia um conjunto de funcionalidades como:

- Gestão da documentação;
- Gestão de tarefas (planeamento e controlo);
- Mecanismos de comunicação e notificação internos;
- Entre outros.

A experiência adquirida com o sistema evidenciou que a aplicação não satisfaz as necessidades referidas anteriormente. Para tal, é necessário desenvolver dois módulos:

- Gestor flexível de processos;
- Gestor configurável de conteúdos provenientes de diferentes suportes;

Os capítulos seguintes são dedicados à especificação, concepção e implementação destes dois módulos.

Capítulo 4

Gestor de processos

Neste capítulo será efectuada uma abordagem ao módulo de gestão de processos, referido no Capítulo 3.3, desde a sua concepção até à sua implementação.

Começa-se na Secção 4.1 por mencionar os requisitos que este módulo deve satisfazer. De seguida, na Secção 4.2 são apresentados alguns aspectos de concepção deste módulo, nomeadamente a arquitectura geral, o modelo de dados e a interface idealizada. Por último, na Secção 4.3, é apresentada a forma como foi efectuada o acesso a dados e interface final deste módulo.

4.1 Análise de requisitos

No capítulo anterior, através da análise do sistema de trabalho dos STM, verificou-se que existe uma grande diversidade de processos, com diferentes níveis de estruturação e sem um fluxo rígido de execução (processos semi-estruturados). O gestor flexível de processos surgiu com o objectivo de dar suporte a processos com este cariz e, consequentemente, a sistemas de trabalho com as características do sistema de trabalho dos STM. Para tal, foram identificados um conjunto de requisitos (ver Tabela 4.1) que deve satisfazer para que consiga cumprir o seu objectivo.

Para que um processo semi-estruturado possa ser gerido num sistema informático de suporte, deve ser previamente elaborado um modelo (Req.1) mais ou menos estruturado, dependendo das características do processo. Esse modelo é constituído por um conjunto de fases (Req.4), que podem conter várias tarefas(Req.5). Consoante o grau de estruturação do processo, pode ser pertinente no modelo do processo definir-se precedências entre tarefas (Req.8) e tarefas obrigatórias ou não obrigatórias (Req.7), tal como acontece, alias, no DECLARE e no UAM.

É muito frequente ser necessário guardar-se informações adicionais numa tarefa, e.g., custo da reparação. Logo, o gestor de processos deve permitir associar atributos de diferentes formatos a uma determinada tarefa (Req.6), para que cada tarefa tenha os atributos adicionais mais convenientes ao seu suporte, sem que fique informação por associar a uma tarefa.

Nº	Requisito
1	Deve ser possível a criação de modelos de processos
2	Deve ser possível a edição de modelos de processos
3	Aquando a criação de um modelo, deve ser possível associá-la a diferentes unidades da organização
4	Deve ser possível associar-se fases a um modelo do processo
5	Deve ser possível associar-se tarefas a uma fase
6	Deve ser possível associar-se dados (atributos) complementares a uma tarefa da instância
7	Deve ser possível definir-se tarefas obrigatórias e não obrigatórias no modelo do processo
8	Deve ser possível especificar-se precedências entre as tarefas no modelo do processo
9	Deve ser possível criar-se instâncias do processo referentes a um modelo do processo
10	Deve ser possível editar-se instâncias do processo
11	Aquando a criação de uma instância, deve ser possível associá-la a diferentes unidades da organização
12	O controlo da execução das tarefas da instância do processo deve ser efectuado com recurso a uma <i>checklist</i>
13	Deve ser possível colocar-se uma tarefa da instância do processo como não aplicável
14	Deve ser possível editar-se os dados dos atributos de uma tarefa aquando a gestão do processo
15	Deve ser possível para cada unidade saber os seus processos e o estado dos mesmos

Tabela 4.1: Requisitos do módulo de gestão flexível de processos

Tipicamente as organizações estão organizadas por unidades hierarquizadas. Assim, aquando a criação de um modelo do processo, deve-se associá-lo a uma unidade à qual estará disponível para futuras instanciações (Req.3).

Um modelo de um processo frequentemente evolui no tempo em função da experiência adquirida na gestão do processo. Por isso, deve ser possível editar-se o modelo após a sua criação. Assim, os dados intrínsecos ao modelo, tal como, dados gerais (nome, unidade a que pertence, etc.), fases, tarefas, e atributos devem poder ser alterados (Req.2).

Após estar definido o modelo do processo, deve haver possibilidade de criar-se instâncias desse modelo (Req.9). Como a instância é referente a um modelo, após ser criada terá disponível todas as tarefas e fases que foram definidas no modelo. Assim, é necessário algum mecanismo que permita ao utilizador gerir a execução das tarefas da instância do processo. Nesta dissertação, tal como no UAM, optou-se pela *checklist* para executar o controlo da execução das tarefas (Req.12), por ser um mecanismo flexível ao nível do controlo do trabalho realizado.

A abordagem da *checklist* é completamente diferente em relação à utilizada nos sistemas de

workflow. Nestes, cada utilizador só tem acesso a visualizar a tarefa que precisa de realizar e perde inclusivamente uma imagem do processo como um todo, por se focar apenas nas suas tarefas. Isto é possível por se tratarem de processos rigidamente definidos ao nível do fluxo de execução. Mas em processos semi-estruturados, como tipicamente envolvem conhecimento e estão dependentes da arte de quem os executa, é extremamente complicado arranjar um sistema que deduza qual a melhor tarefa a realizar, porque isso efectivamente é uma decisão que parte de quem o estiver a realizar. Uma melhor abordagem nestes casos é a *checklist*, por disponibilizar as tarefas e o seu estado como uma referência à execução do processo, dando assim liberdade para o utilizador decidir o rumo que deve dar ao processo. Assim, em contraste com os sistemas de workflow, numa *checklist* pode-se visualizar todas as tarefas do processo (trabalho a realizar) e o seu estado (realizada ou não realizada), tendo-se assim uma perspectiva geral do processo que permite tomar decisões em relação ao melhor caminho que este deve seguir, sem condicionar as opções de quem o executa.

A instância do processo, com o desenrolar dos trabalhos, vai evoluindo. Logo, é importante que seja possível editá-la (Req.10) ao nível dos seus dados gerais (título, descrição, etc.), do estado das suas tarefas na *checklist* e ao nível dos dados dos atributos pertencentes a uma determinada tarefa (Req.14).

Aquando da criação de uma instância, deve-se definir a que unidade pertence (Req.11). Embora o modelo do processo esteja associado a uma unidade da organização e as instâncias dos processos, por sua vez, dizerem respeito a um modelo do processo, é necessário ainda que a instância também se encontre associada a uma unidade, uma vez que um modelo está disponível para uma unidade e consequentemente para todas as unidades que se encontrem a um nível inferior na hierarquia. Isto é importante para posteriormente gerar-se uma visão consolidada dos processos de uma determinada unidade, porque as instâncias dos processos podem ficar associadas a uma unidade diferente da do seu modelo.

É importante gerar uma vista consolidada do estado dos processos de uma determinada área, que permita o gestor do serviço ter noção do estado da sua unidade (Req. 15).

4.2 Concepção

Nesta secção, será concebida a aplicação que dará resposta aos requisitos mencionados na secção anterior. Assim, em primeiro lugar, será apresentada a arquitectura geral da aplicação. De seguida, dar-se-á mais ênfase ao modelo de dados da aplicação e às interfaces da mesma.

4.2.1 Arquitectura

Perante os requisitos mencionados na secção anterior, foi possível definir-se a arquitectura geral da Figura 4.1. Através da figura é possível visualizar as camadas que compõe esta arquitectura, os módulos de cada camada e as respectivas interacções entre eles.

As três camadas da arquitectura são:

- **Camada de apresentação:** responsável por interagir directamente com o utilizador;
- **Camada lógica:** contém todas as regras e funções responsáveis pela lógica da aplicação; trata os pedidos provenientes da camada de apresentação, invocando, se necessário, a camada de dados;
- **Camada de acesso a dados:** recebe os pedidos da camada lógica e executa-os na base de dados.

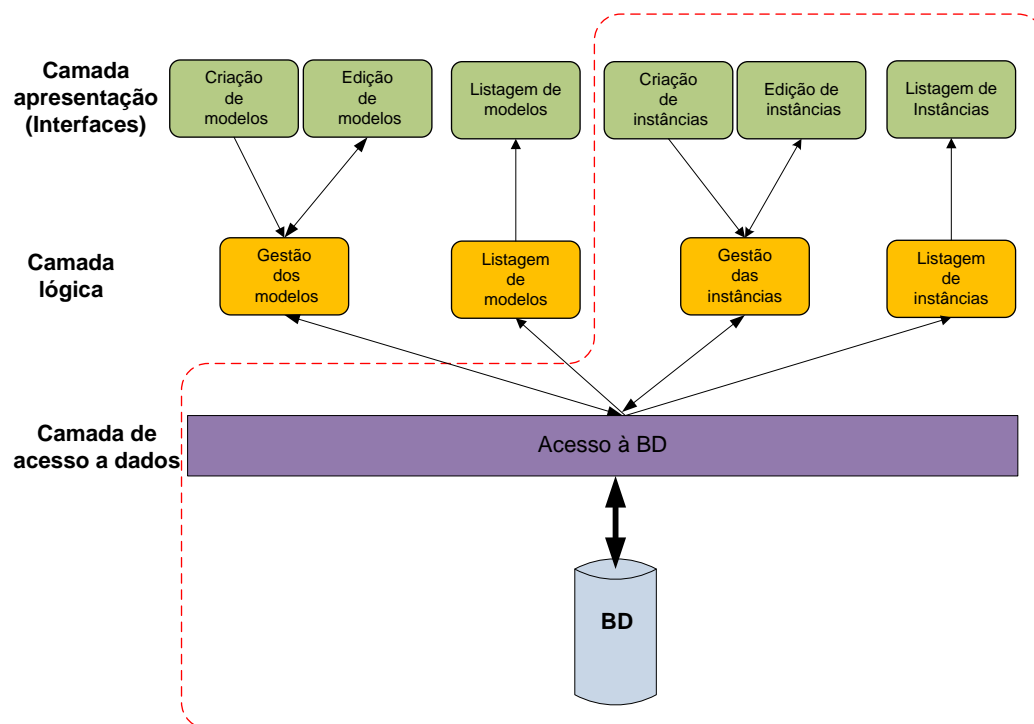


Figura 4.1: Arquitectura geral do gestor de processos

A divisão em três camadas lógicas possibilita futuras alterações numa camada sem prejudicar as demais camadas. Por exemplo, alterando-se a base de dados bastaria alterar-se posteriormente a camada de acesso a dados, porque as restantes camadas não seriam afectadas por essa alteração.

A camada de apresentação é composta pelas seguintes interfaces:

- **Criação de modelos:** permite ao utilizador criar um modelo do processo novo, definindo as tarefas do processo, as fases, regras e etc.;
- **Edição de modelos:** permite ao utilizador editar o modelo de um processo já existente;
- **Listagem de modelos:** permite ao utilizador visualizar os modelos que estão disponíveis na páginas de uma unidade;
- **Criação de instâncias:** permite ao utilizador criar uma instância que tenha por base um modelo do processo previamente definido.

- **Edição de instâncias:** permite ao utilizador editar uma instância já existente, possibilitando assim a gestão do processo ao longo do seu ciclo de vida.
- **Listagem de instâncias:** permite ao utilizador visualizar as instâncias de processos existentes na páginas de uma unidade;

A camada lógica é constituída por três módulos que suportam as interfaces anteriormente mencionadas. Esta camada faz pedidos à camada de acesso a dados e trata a informação de forma a que possa ser apresentada ao utilizador na camada superior (camada de apresentação).

A camada de acesso a dados será composta por uma classe que ficará encarregue de disponibilizar todos os métodos de acesso à base de dados, para que possam ser invocados pela camada superior.

A arquitectura do gestor de processos (Figura 4.1) é composta por módulos que dizem respeito aos modelos e às instâncias, ao nível da criação, edição e listagem. Como neste trabalho se pretende desenvolver um protótipo de um sistema de suporte de processos semi-estruturados e como o tempo para a implementação desse sistema não é muito, decidiu-se desenvolver apenas os módulos mais importantes do gestor de processos (módulos que se encontram dentro da caixa a vermelho na arquitectura), que são os módulos que dizem respeito às instâncias do processo. Mas como para gerir uma instância do processo é necessário o seu modelo, os modelos terão de ser inseridos, por enquanto, directamente na BD, uma vez que, não serão desenvolvidas as interfaces nem o módulo da camada lógica para o fazer. Contudo, através dos módulos desenvolvidos já dá para testar a forma como o gestor de processos permite aos actores gerirem os processos com o seu auxílio. Posto isto, nas secções seguintes não serão abordados os módulos que dizem respeito aos modelos, porque eles efectivamente não são alvo de implementação.

4.2.2 Modelo de dados

O modelo de dados que suporta os requisitos deste módulo, foi dividido em três pacotes lógicos de dados relacionados entre si, sendo eles:

- **Modelos dos processos:** pacote que contém todas as entidades que dizem respeito à modelação de processos;
- **Instâncias dos processos:** pacote que contém todas as entidades que dizem respeito às instâncias de processos;
- **Navegação:** pacote que contém todas as entidades que dizem respeito às unidades da organização, às quais estarão associadas as restantes entidades;

De seguida, será efectuada uma análise mais pormenorizada de cada um dos pacotes mencionados. Posteriormente será apresentado o modelo de dados completo, onde estão os três pacotes e as interacções entre eles.

4.2.2.1 Modelos dos processos

Este pacote de dados contém todas as entidades e associações necessárias para suportar todos os requisitos relacionados com os modelos dos processos (Req.1 e 2 e Req.4 a 8).

Na Figura 4.2 está presente o modelo de dados deste pacote. As entidades que o compõem, são:

- Modelo do Processo
- Fase
- Tarefa
- Atributo
- TipoAtributo

Um modelo do processo pode ter várias fases e, por sua vez, uma fase pode ter várias tarefas. Foi colocada a relação de N para N entre as tarefas para permitir estabelecer precedências entre as mesmas (Req.8). Esta relação não podia ser de 1 para N, porque uma tarefa pode ter várias precedentes. O atributo "obrigatoria" permite definir se é obrigatório executar ou não uma tarefa. Já o "ordem", serve para guardar a ordem em que as tarefas aparecerão na *checklist*, tendo apenas impacto na disposição das tarefas na interface.

A relação de 1 para N entre as Fases do modelo do processo foi colocada para estabelecer uma ordem entre as fases. Como iremos ver, esta relação será importante para determinar o estado do processo.

A entidade Atributo tem como objectivo permitir ao utilizador associar atributos às tarefas. Para isso é necessário saber o nome do atributo a apresentar ao utilizador, p.ex. "Sexo". Contudo, a natureza dos atributos pode ser variável. O utilizador tanto pode querer que uma tarefa tenha um atributo que dê para inserir texto simples, como um atributo que permita escolher um valor de uma lista, como está na Figura 4.3. Para tal, é necessário guardar no campo "parametros" da entidade Atributo quais serão os valores a mostrar ao utilizador. No caso do exemplo da figura, o campo parâmetro poderia ter o seguinte valor: "masculino;feminino". Com esta informação e sabendo qual o controlo a apresentar na interface para que o utilizador possa utilizar o atributo, através do campo "controlo" da entidade TipoAtributo, é possível deixar o utilizador associar atributos dinamicamente às tarefas dos modelos dos processos, para que mais tarde possa usá-los na gestão do processo.

4.2.2.2 Instâncias dos processos

Este pacote de dados contém todas as entidades e associações necessárias para suportar todos os requisitos relacionados com os modelos dos processos (Req.9 e 10 e Req.12 a 14).

Na Figura 4.4 está presente o modelo de dados deste pacote. As entidades que o compõem, são:

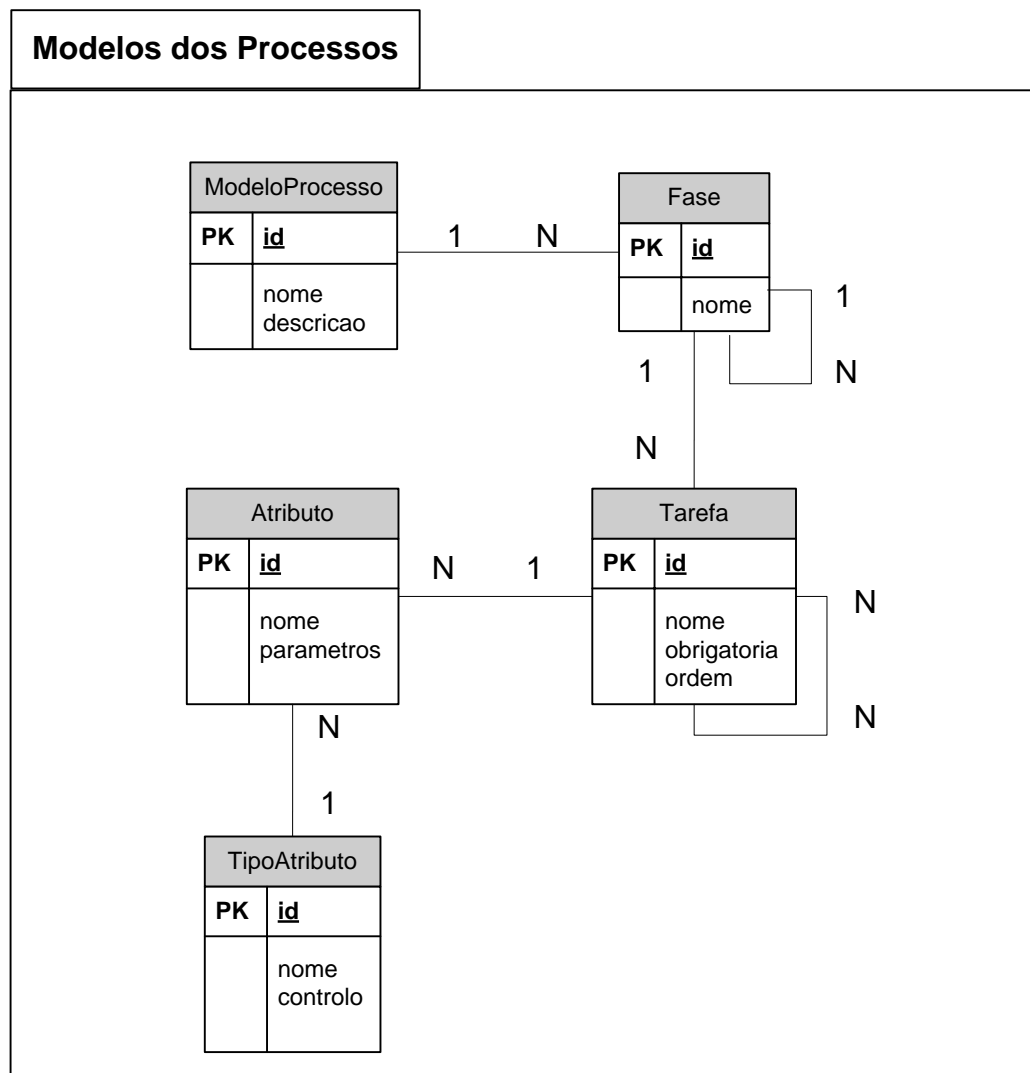


Figura 4.2: Modelo de dados do pacote modelos dos processos

Sexo: ☐ Masculino ☐ Feminino

Figura 4.3: Exemplo de atributo a associar a uma tarefa

- InstanciaProcesso
- EstadoProcesso
- TarefaRealizada

- ValorAtributo

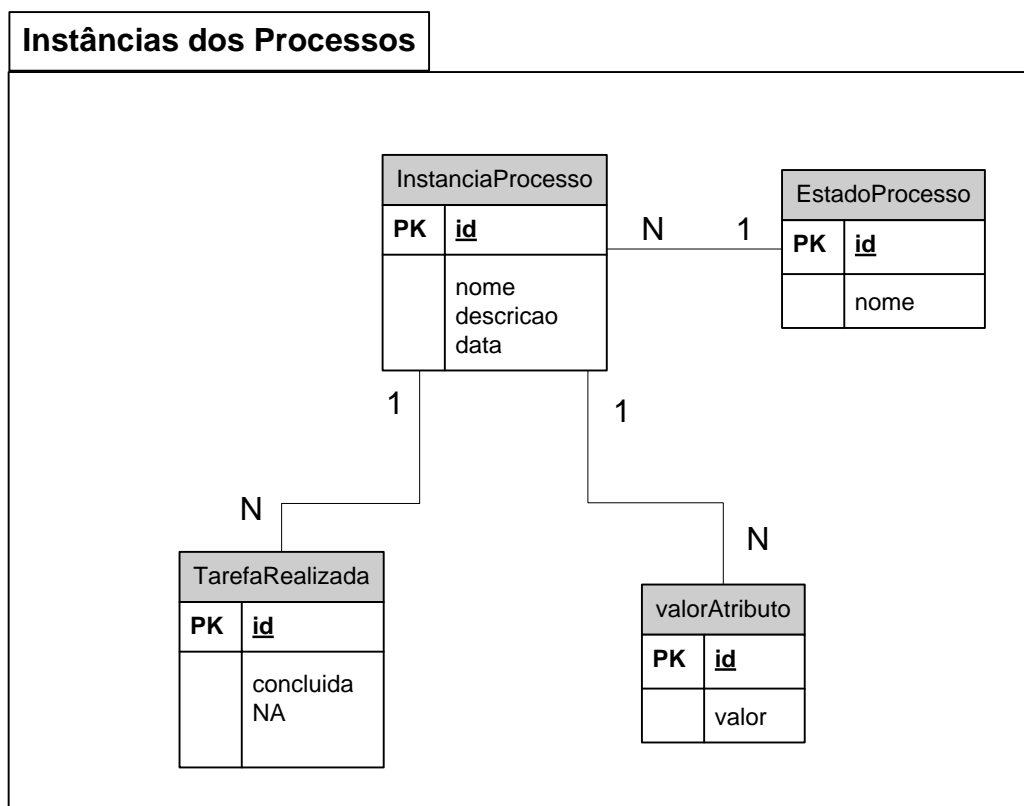


Figura 4.4: Modelo de dados do pacote instâncias dos processos

Uma instância de um processo, possui um estado (Em Curso, Arquivada, Planeado, etc..). Daí existir uma relação de 1 para N entre a entidade InstanciaProcesso e EstadoProcesso.

Uma instância de um processo é relativa a um modelo. Como um modelo possui várias tarefas, é necessário, ao longo da gestão do processo, guardar o estado (concluída ou não concluída) das tarefas da instância. Isso pode ser guardado no campo "concluída" da entidade TarefaRealizada. O mesmo acontece com os atributos de cada tarefa, em que o seu valor pode ser guardado no campo "valor" da entidade ValorAtributo.

4.2.2.3 Navegação

O pacote Navegação surge com o objectivo de dar resposta aos Requisitos 3 e 11. Tipicamente as organizações encontram-se organizadas por departamentos ou áreas, de uma forma hierarquizada. Isto torna necessário que aquando da criação de um processo ou de um modelo do processo se associe o mesmo à área à qual ele pertence. Para isso, no modelo de dados tem de estar presente a estrutura da organização.

Pegando na organização dos STM, como vimos no Capítulo 3.1, os STM possuem várias unidades e nelas são geridos equipamentos/recursos com semelhanças. Contudo, por vezes

esses recursos podem ser subdivididos de forma a organizar o trabalho dentro das unidades. Na Figura 4.5 está um exemplo disso mesmo, em que as unidades dos STM são divididas por famílias de equipamentos, por sua vez, estas podem ser divididas por tipo de equipamento e, por ultimo, nestas estão instâncias de equipamentos.

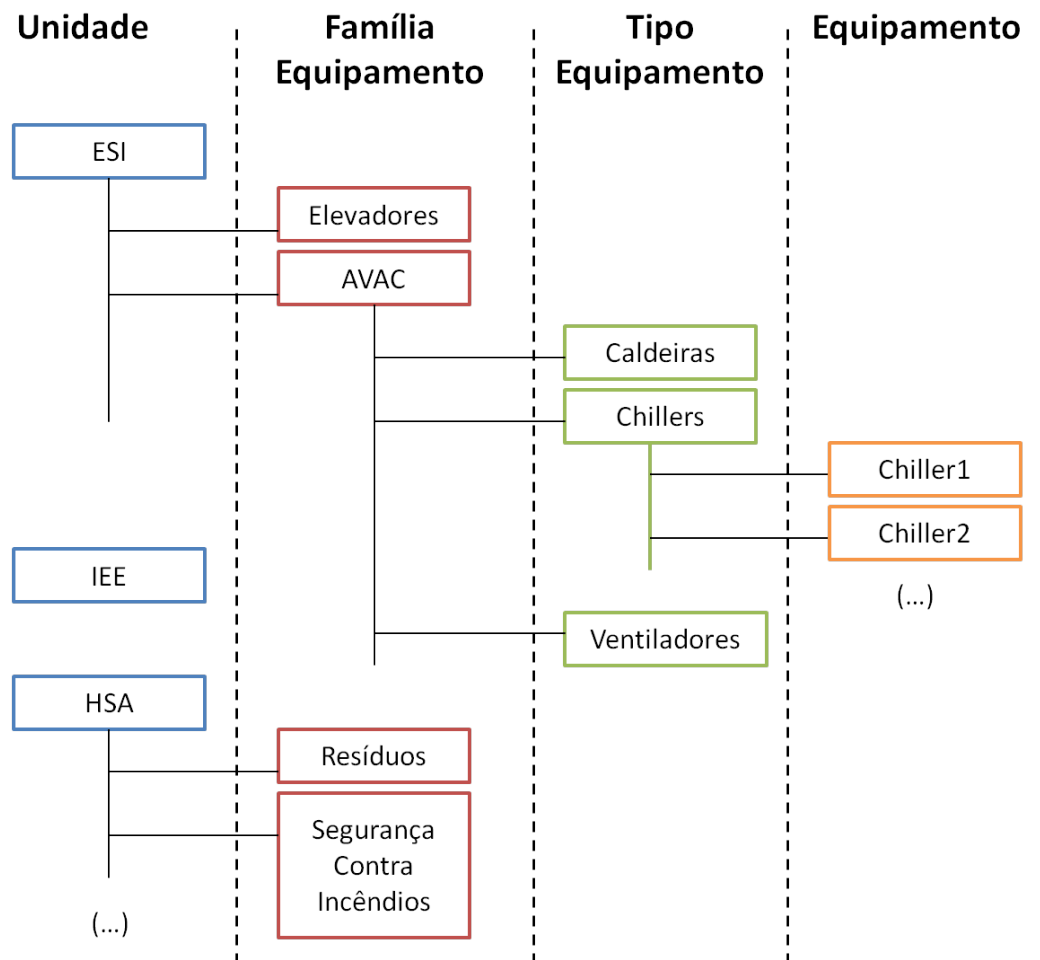


Figura 4.5: Organização hierarquizada dos STM

O modelo de dados deste pacote (ver Figura 4.6) permite então armazenar a estrutura da organização, para que posteriormente se possa associar outras entidades às varias áreas da organização. As entidades que o compõem são:

- AreaActividade;
- Unidade;
- Família;
- TipoEquipamento;
- Equipamento.

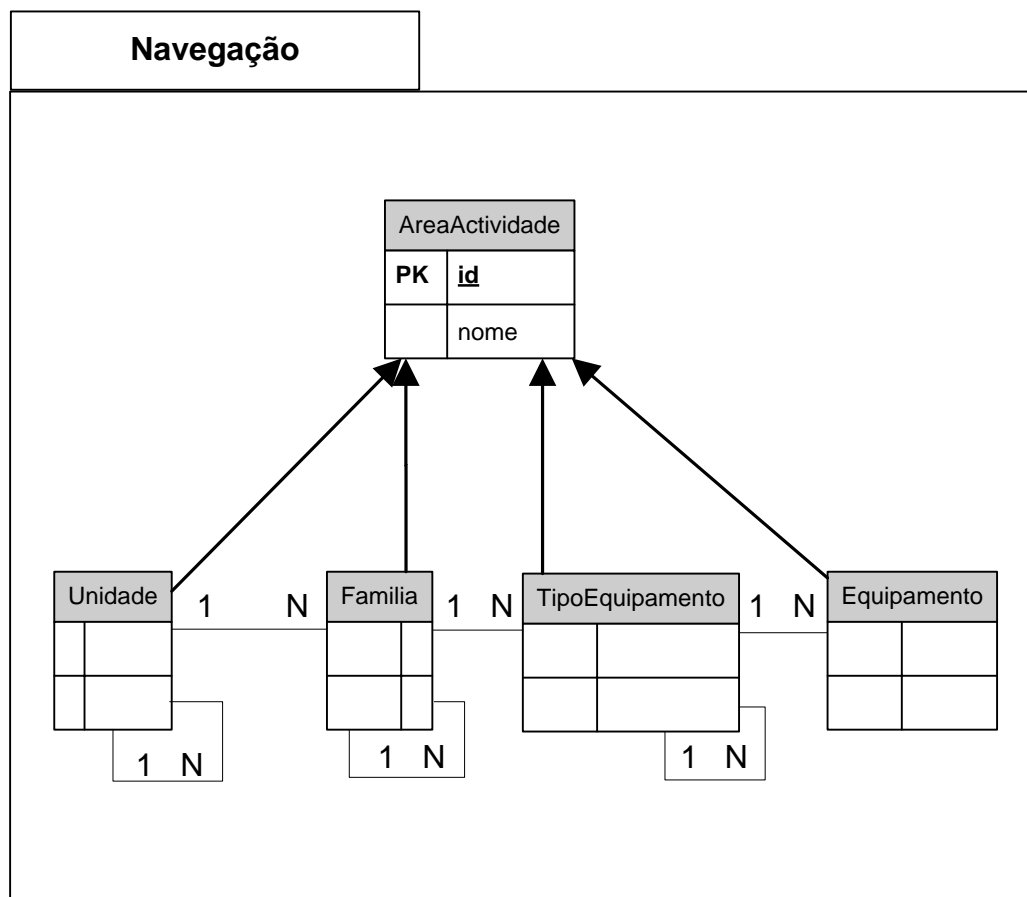


Figura 4.6: Modelo de dados do pacote navegação

A entidade *AreaActividade* guardará todas as áreas da organização. As restantes entidades serão subconjuntos da informação presente na entidade *AreaActividade*, de forma a que seja possível distinguir se uma área de actividade é uma unidade, uma família de equipamento, um tipo de equipamento ou um equipamento.

As relações de 1 para N entre as entidades *Unidade*, *Familia*, *TipoEquipamento* e *Equipamento* têm como objectivo construir a hierarquia entre áreas de actividade, para que seja possível colocar no sistema que, por exemplo, a família de equipamento AVAC pertence à unidade ESI (tal como na Figura 4.5).

As relações de 1 para N existentes nas entidades *Unidade*, *Família* e *TipoEquipamento* servem para inserir vários níveis hierárquicos na mesma entidade, ou seja, desta forma o ESI pode ter uma sub-unidade a um nível inferior, para além de uma família de equipamento. Isto permite a existência de sub-unidades, sub-famílias e sub-tipos de equipamentos.

A navegação poderia ser feita apenas com a entidade *AreaActividade* e uma relação de 1 para N na própria entidade. Mas tal abordagem, inviabilizaria uma futura associação de outras entidades a unidades, famílias de equipamentos, tipos de equipamentos e equipamentos, porque estas unidades

não existiriam no modelo de dados. Esta abordagem seria boa para posteriormente criar-se a árvore de navegação da organização, uma vez que toda a informação estaria contida na entidade *AreaActividade*. Mas com a abordagem adoptada, tal também poderá ser realizado, se se efectuar uma vista na base de dados que agregue a informação existente em todas as entidades do pacote, tendo-se assim uma vista com as relações entre todos as unidades, famílias de equipamentos, tipos de equipamentos e equipamentos.

4.2.2.4 Modelo completo

Até ao momento descreveu-se cada pacote de dados isoladamente. Mas acontece que existem várias interacções entre os pacotes. O modelo de dados completo do gestor de processos e as interacções entre pacotes podem ser visualizadas através da Figura 4.7.

O pacote Modelos dos Processos interage com o Instâncias dos Processos através das seguintes relações de 1 para N:

- **entre ModeloProcesso e InstanciaProcesso:** para que uma instância do processo fique associada ao modelo ao qual pertence;
- **entre Tarefa e TarefaRealizada:** para associar uma tarefa realizada na instância do processo à tarefa correspondente no modelo;
- **entre Atributo e ValorAtributo:** para que se saiba a que atributo do modelo do processo pertence um valor inserido na instância.

As interacções entre os pacotes Modelos dos Processos e Instâncias dos Processos com o pacote Navegação, surgem pela necessidade de:

- Para cada área de actividade listar-se os modelos que a ela estão associados (daí a relação 1 para N entre *AreaActividade* e *ModeloProcesso*);
- Para cada área de actividade listar-se as instâncias que a ela estão associadas (daí a relação 1 para N entre *AreaActividade* e *InstanciaProcesso*).

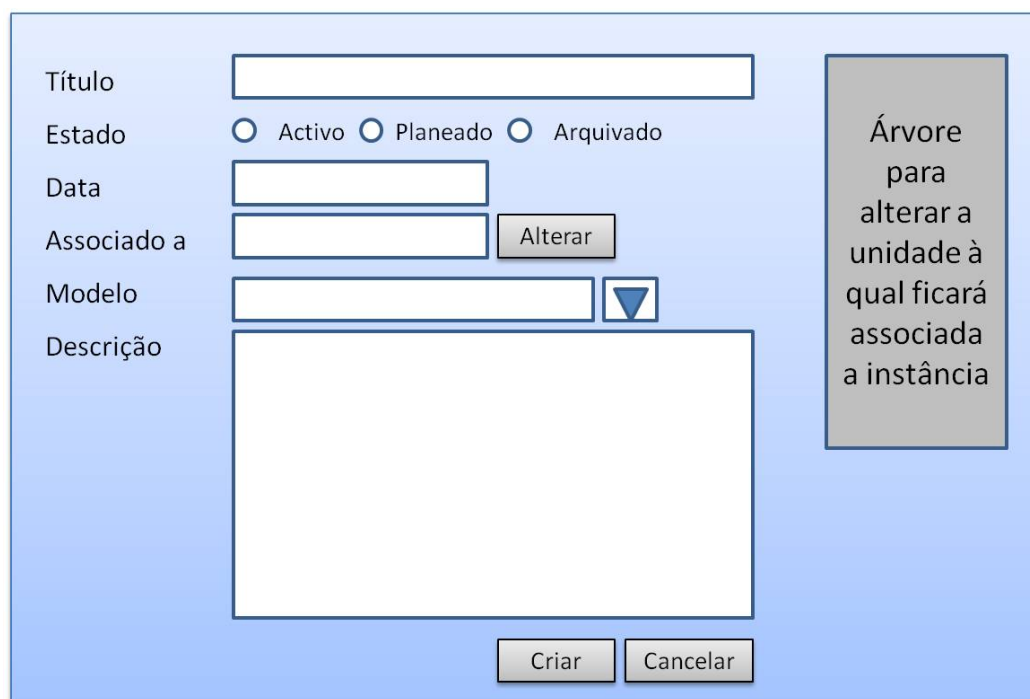
Estas duas relações são necessárias, porque um modelo de processo quando é associado a uma área de actividade fica disponível para essa área e para todas a um nível inferior. Por isso, quando se cria uma instância do processo, é necessário especificar-se a que área ela fica associada para que posteriormente apareça listada na área correcta.

4.2.3 Interfaces

Nesta secção será ilustrado um esboço das interfaces a implementar e uma pequena descrição das mesmas.

4.2.3.1 Criação de Instâncias

O esboço da interface idealizada para a criação de instâncias está presente na Figura 4.8.



O esboço da interface de criação de instâncias de processos apresenta os seguintes elementos:

- Título:** Campo de texto para o título da instância.
- Estado:** Três botões de opção: ☐ Activo, ☐ Planeado, ☐ Arquivado.
- Data:** Campo de texto para a data.
- Associado a:** Campo de texto para a unidade associada, acompanhado de um botão "Alterar".
- Modelo:** Campo de texto para o modelo, acompanhado de um botão de seta para baixo.
- Descrição:** Área de texto grande para a descrição da instância.
- Botões de Ação:** Botões "Criar" e "Cancelar" na base da interface.
- Árvore de Unidades:** Uma caixa cinzenta à direita com o texto "Árvore para alterar a unidade à qual ficará associada a instância".

Figura 4.8: Esboço da interface de criação de instâncias de processos

O campo "Associado a" deve ser automaticamente preenchido quando abrir esta interface, em função da área de actividade em que a aplicação se encontrava anteriormente. Contudo, tal poderá ser alterado através do botão Alterar que se encontra ao lado. Caso se carregue neste botão, deverá aparecer (na caixa cinzento ao lado) uma árvore que permita ao utilizador escolher a que área de actividade pretende associar a instância. Em função desta escolha, aparecerá no campo Modelos, os modelos disponíveis para instanciar na área seleccionada.

Assim, após o utilizador preencher todos os campos, escolher a que área pretende associar a instância e definir qual o modelo que pretende, poderá carregar no botão Criar e posteriormente a instância será criada no modelo de dados.

4.2.3.2 Listagem de Instâncias

O esboço da interface idealizada para a listagem de instâncias está presente na Figura 4.9.

Estado Processo: ▼

Modelo Processo: ▼ [Novo](#)

Título Processo	Estado
Reparação Chillers	Planeamento Orçamento
Reparação AVAC	Concluído

Figura 4.9: Esboço da interface de listagem de instâncias de processos numa área de actividade

Esta interface deverá para uma dada área de actividade listar os processos que lhe pertencem. Essa listagem deve poder ser filtrada através do estado do processo (Activo, Planeado, Arquivado) ou através dos modelos dos processos das instâncias que pertencem à área de actividade em questão e satisfazem o estado do processo seleccionado.

A lista de instâncias de processos deve permite aceder à edição da instância (interface descrita na secção seguinte) ao carregar-se no título do processo e também deve permite ao gestor da área de actividade saber o estado dos processos da mesma, tendo assim uma vista agregada de todos os processos que estão sobre a sua gestão e o seu estado.

Através do botão "Novo" deve ainda ser possível aceder-se à interface de criação de uma nova instância do processo (interface descrita na secção anterior).

4.2.3.3 Edição de Instâncias

O esboço da interface idealizada para a edição de instâncias está presente na Figura 4.10.

Esta interface é a mais complexa do gestor de processos. Para além permitir a edição dos dados da instância do processo, tem de possibilitar a gestão do processo ao longo do seu ciclo de vida. Por isso, pode ser vista como contendo duas partes distintas, uma para edição dos dados da instância e outra para a gestão do processo. Estas duas partes estão na mesma interface para tornar toda a informação do processo o mais acessível possível para o utilizador.

Na parte superior, encontra-se a parte de edição dos dados onde deve ser possível editar alguns dos dados da instância. A área à qual se encontra associada e o seu modelo não são campos editáveis, apenas se encontram na interface para o utilizador ter acesso a essa informação. Após

Título:

Estado: ☒ Activo ☐ Planeado ☐ Arquivado

Data:

Associado a:

Modelo:

Descrição:

Gestão do Processo

☒ **Fase1**

- ☐ Tarefa1
- ☐ Tarefa2
- ☒ Tarefa3
- ☒ Não aplicáveis
- ☐ Tarefa4

☒ **Fase2**

☒ **Fase3**

- ☐ Tarefa5
- ☐ Tarefa6

Não Aplicável ☐

Notas:

Fornecedor: ▼

Figura 4.10: Esboço da interface de edição de instâncias de processos

o utilizador alterar os dados que pretende nesta parte e carregar no botão Alterar, as alterações devem ser efectuadas na BD e a aplicação deve ser encaminhada para a página da unidade à qual pertence o processo.

Na parte inferior à anterior, encontra-se a parte de gestão do processo. Nela deve de haver a possibilidade de visualizar-se as tarefas e as fases do processo, colocar tarefas como realizadas ou não realizadas na *checklist*, inserir dados adicionais nas tarefas e colocar tarefas como não

aplicáveis.

Se o utilizador carregar no nome de uma tarefa, deve abrir a janela ao lado com a informação adicional da tarefa. Na Figura 4.10 a tarefa que tem a janela aberta é a "Tarefa1". Desta forma, é possível alterar os campos pertencentes à tarefa. É de salientar que todas as tarefas têm o campo "Não aplicável" por defeito, mas os restantes campos são os que foram associados à tarefa aquando a criação do modelo. Desta forma, para cada tarefa pode-se guardar dados diferentes em função daquilo que é necessário ficar registado na sua execução. Após o utilizador alterar os dados que pretende de uma tarefa, se carregar em Guardar esses dados devem ser imediatamente guardados na base de dados e a janela da tarefa deve fechar, podendo de seguida carregar noutra tarefa e abrir de novo a janela, mas para a tarefa que carregou.

As tarefas que são colocadas como não aplicáveis, são "arrumadas" dentro da fase a que pertencem, por baixo do tema "Não Aplicáveis". Assim, as tarefas que não se aplicam à instância em questão ficam colocadas separadamente das que se aplicam, não confundindo o utilizador. Nas tarefas que não se aplicam, não é possível colocar o visto na tarefa.

As alterações efectuadas na *checklist*, devem surtir efeito de imediato na base de dados. Por isso é que não existe um botão para confirmar essas alterações.

4.3 Implementação

Conforme foi referido no Capítulo de introdução, os módulos desenvolvidos destinavam-se a ser integrados na aplicação de gestão dos STM, desenvolvida em SP. A forma mais comum de se acrescentar funcionalidades ao SharePoint é através da construção de Peças Web [8]. No SP, as páginas são constituídas por uma ou mais Peças Web, sendo esta a unidade de informação modular que serve de base para a construção visual das referidas páginas.

De forma a inserir no sistema dos STM o gestor de processos, foram desenvolvidas três Peças Web em C# e .NET, cada uma contendo as três interfaces mencionadas na secção anterior.

A implementação do módulo de gestão de processos é relativamente complexa e não seria possível apresentar neste documento uma análise detalhada deste módulo. Assim, será dada ênfase à camada de acesso a dados, a qual foi implementada com recurso ao LINQ. O capítulo concluirá com a apresentação das interfaces, tal como elas são apresentadas ao utilizador.

4.3.1 Acesso a Dados

Para a implementação da camada de acesso a dados foi utilizado o Language Integrated Query (LINQ). De seguida, será efectuada uma pequena descrição desta tecnologia, tendo como base [3, 2].

O LINQ consiste num conjunto de recursos introduzidos no .NET Framework 3.5 que permitem a realização de consultas directamente em estruturas de dados (LINQ to Objects), na base de dados (LINQ to SQL) e em documentos XML (LINQ to XML), usando uma sintaxe parecida com o SQL e não obrigando o utilizador a deixar o conforto das linguagens de programação como o C# e o Visual Basic.

A forma mais fácil de perceber o LINQ é começar-se por analisar um exemplo de LINQ para objectos. Por exemplo, se pretendêssemos uma lista de empregados que têm o último nome começado pela letra D, usando código C# teríamos de navegar por toda a colecção de empregados e guardar aqueles que tivessem o ultimo nome começado por D. Isso poderia ser feito através do código da Figura 4.11.

```
//obtenção dos funcionários através de um método auxiliar
List<EmployeeDetails> employees = db.GetEmployees();

//procura dos funcionários que tem o último nome começado pela letra D
List<EmployeeDetails> matches = new List<EmployeeDetails>();

foreach (EmployeeDetails employee in employees)
{
    if (employee.LastName.StartsWith("D"))
    {
        matches.Add(employee);
    }
}
```

Figura 4.11: Criação da lista de empregados usando uma abordagem tradicional em C#

Esta tarefa poderia ser realizada usando uma expressão LINQ em C#, que o resultado seria o mesmo. A Figura 4.12 ilustra como o código anterior pode ser reescrito, substituindo a expressão *foreach* por uma query LINQ. Se analisarmos este código, reparamos que estamos a usar expressões parecidas com SQL para filtrar numa lista os empregados que interessam, usando para isso código apenas C#.

```
List<EmployeeDetails> employees = db.GetEmployees();
IEnumerable<EmployeeDetails> matches;

matches = from employee in employees
           where employee.LastName.StartsWith("D")
           select employee;
```

Figura 4.12: Criação da lista de empregados usando LINQ

Uma das partes mais interessantes do LINQ é o LINQ para SQL. Este permite o uso de expressões LINQ (como a que vimos anteriormente) para efectuar *queries* na base de dados, sendo estas posteriormente traduzidas em SQL puro. Para além de o LINQ para SQL permitir consultas na base de dados, também permite apagar e actualizar dados.

Quando se faz uma *query* à base de dados com LINQ para SQL, os resultados desta provenientes da base de dados são colocados num grupo de objectos em memória, fazendo-se assim uma tradução da informação que está na base de dados para um conjunto de objectos. Para tal, é necessário criar um conjunto de classes que correspondam às entidades que se encontram na base de dados. Essas classes podem ser efectuadas manualmente, ou geradas automaticamente através do Visual Studio.

Uma vez criadas as classes de dados correspondentes às entidades existentes na base de dados, está tudo apto para se usar LINQ para SQL. Neste módulo, foi criada uma classe para cada entidade referida no modelo de dados, sendo assim possível efectuar-se *queries* à base de dados através do LINQ para SQL. Cada classe criada tem os métodos necessários para interagir com a tabela correspondente na base de dados, sendo depois esses métodos invocados pelos blocos da camada lógica.

A Figura 4.13 ilustra um função pertencente à classe *InstanciaProcesso*, que corresponde à representação da entidade *InstanciaProcesso* do modelo de dados. Esta função recebe o *id* do processo a pesquisar na base de dados, executa uma query LINQ à base de dados e posteriormente coloca o resultado num objecto do tipo *InstanciaProcesso*, que contém todos os atributos que estão na base de dados para a instância de processo em questão. Isto se fosse implementado sem LINQ, era necessário efectuar-se uma ligação à base de dados, depois executar a query, e por último fechar-se a ligação à base de dados. Entretanto, teria sido necessário alterar-se da linguagem de programação utilizada, para SQL de modo a interagir com a base de dados. Usando LINQ para SQL, não é necessário preocupações em relação à ligação à base de dados, nem é preciso saltar-se de SQL para outra linguagem. No exemplo da figura, o código está todo escrito em C#.

```
public static InstProcesso GetById(BaseDadosSTMDDataContext _db, int _idProcesso)
{
    var _inst = from _i in _db.InstProcessos
                where _i.id == _idProcesso
                select _i;

    return _inst.Single();
}
```

Figura 4.13: Função que retorna uma instância de processo usando LINQ para SQL

Outra característica muito interessante é a forma como permite fazer *updates* na base de dados. Na Figura 4.14 está um exemplo de uma função que permite realizar *update* a uma instância de processo. É necessário obter-se em primeiro lugar o objecto que se pretende alterar, no exemplo da figura invocou-se a função anterior porque ela retorna a instância de um processo pelo seu *id*. Depois de se obter o objecto, pode-se alterar os seus atributos e para que as alterações efectuadas ao objecto sejam gravadas na base de dados, basta invocar-se a função "SubmitChanges". Como dá para ver pelo excerto de código, os *updates* são extremamente simples de ser realizados com LINQ.

```
public static void SetById(BaseDadosSTMDDataContext _db, int _idProcesso,
                          string _descricao, DateTime _date)
{
    InstProcesso _instProcesso = GetById(_db, _idProcesso);

    _instProcesso.descricao = _descricao;
    _instProcesso.data = _date;

    _db.SubmitChanges();
}
```

Figura 4.14: Função que permite fazer *update* a uma instância de processo usando LINQ para SQL

A forma como se pode apagar instâncias de processos da base de dados é muito parecida com a forma como se realiza alterações. Primeiro, coloca-se num objecto a instância que se pretende apagar, depois através do comando "DeleteOnSubmit" diz-se que objecto deverá ser apagado e a que classe pertence, e por último chama-se o comando "SubmitChanges" para que a instância do processo seja apagada na base de dados.

```
public static void DeleteById(BaseDadosSTMDDataContext _db, int _idProcesso)
{
    InstProcesso _instProcesso = InstProcesso.GetById(_db, _idProcesso);

    _db.InstProcessos.DeleteOnSubmit(_instProcesso);

    _db.SubmitChanges();
}
```

Figura 4.15: Função que permite apagar uma instância de processo usando LINQ para SQL

4.3.2 Interfaces

De seguida, serão apresentadas as interfaces finais, que suportam este módulo e será efectuada uma descrição das mesmas.

4.3.2.1 Criação de Instâncias

A interface que permite a criação de instâncias do processo está presente na Figura 4.16. Através da mesma, é possível criar-se uma instância de um processo com base num modelo previamente definido e associá-la a uma área de actividade.

O utilizador tem acesso a esta interface por intermédio da interface de Listagem de Instâncias (será abordada de seguida), que se encontra numa área de actividade. Quando esta interface é iniciada, com base no *id* da área de actividade que é passado no URL da página, o campo "Associar a" é preenchido automaticamente para essa área de actividade. Contudo, se o utilizador pretender

Figura 4.16: Interface de criação de instâncias de processos

alterar a área de actividade à qual ficará associada a instância, poderá fazê-lo, carregando no botão com o símbolo de árvore, que se encontra à frente do texto que é automaticamente preenchido inicialmente. Neste caso, aparecerá a árvore que se encontra à direita, onde está presente a estrutura da organização a um nível inferior àquele da área de actividade que deu origem a esta interface. No exemplo da figura, a área de actividade a partir da qual se acedeu a esta interface foi o ESI. Logo, inicialmente apareceu preenchido automaticamente no campo "Associar a" a área de actividade ESI. Contudo, como se carregou no botão com o símbolo de árvore, aparece a árvore da direita, que permite alterar a área de actividade que ficará associada à instância do processo, aparecendo apenas na árvore todas as áreas da actividade que se encontram a um nível hierárquico inferior ao ESI. Carregando-se depois no nó da árvore com o nome, p.ex. "Aquecimento", é actualizado o campo "Associar a" para a nova área de actividade.

No campo Modelo, é possível seleccionar o modelo da instância do processo em função da área de actividade seleccionada previamente.

Após se preencher todos os campos da interface e carregar-se no botão adicionar, a informação será enviada para o módulo Gestão das instâncias da camada lógica, que por sua vez, encaminhará a informação tratada para a camada de acesso a dados, para que a informação seja adicionada à base de dados.

4.3.2.2 Listagem de Instâncias

A interface que permite a listagem de instâncias de processos de uma área de actividade, está presente na Figura 4.17.

Através da mesma, é possível ter-se uma visão consolidada ao nível dos processos de uma área de actividade e aceder às interfaces de criação (abordada na secção anterior) e edição (abordada na próxima secção) de instâncias.

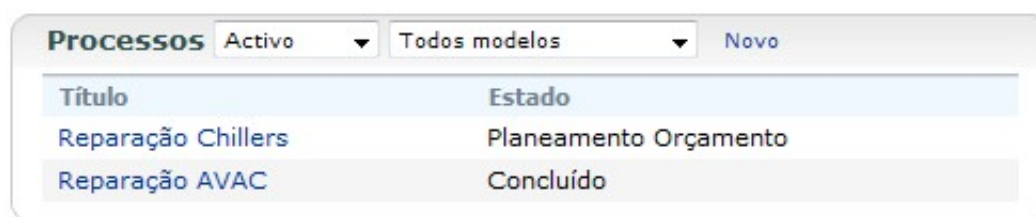
A visão consolidada permite ao gestor da área de actividade saber o estado dos processos que têm sob sua responsabilidade. O estado do processo, consiste na primeira fase do processo que não têm todas as suas tarefas realizadas.

A lista dos processos é obtida com base no *id* da área de actividade que consta no URL da página em que se encontra esta interface. Assim, com base nesse parâmetro e com base na opção seleccionada nos filtros do estado do processo e do modelo, são listadas as instâncias de processos correspondentes.

Após serem listadas as instâncias dos processos, se se carregar no nome das mesmas, a aplicação será reencaminhada para a interface de edição de instâncias e para o processo que foi seleccionado.

O utilizador também poderá carregar no link "Novo", sendo assim reencaminhado para a interface de criação de instâncias. Este link tem um parâmetro que é o *id* da área de actividade que, como vimos na secção anterior, é o responsável por fornecer informação relativa à área de actividade que reencaminhou a aplicação para esta interface.

Área de trabalho: E01 - Elevadores



Processos	
Activo	Todos modelos
Novo	
Título	Estado
Reparação Chillers	Planeamento Orçamento
Reparação AVAC	Concluído

Figura 4.17: Interface de listagem de processos

4.3.2.3 Edição de Instâncias

A interface que permite a edição de instâncias de uma área de actividade, está presente na Figura 4.17.

Com base no *id* do processo passado no URL desta página, é carregada a informação do processo e colocada à disposição do utilizador na interface, para que este possa alterar os dados que constam na parte de edição de dados e na parte de gestão do processo.

Existem campos da parte de edição de dados que não podem ser editáveis, como é o caso da área de actividade à qual está associada a instância do processo e ao modelo ao qual corresponde. Se fosse permitido ao utilizador alterar a área à qual está associada a instância, ele poderia escolher uma área que não dispunha do modelo de processo que foi escolhido aquando a criação da instância. Por isso, optou-se por não deixar alterar estes dois campos.

Editar Processo

Título:

Estado: ☒ Activo ☐ Planeado ☐ Arquivado

Data:

Associado a: ESI > E01 - Elevadores > Elevadores

Modelo:

Descrição:

rep

Gestão do Processo

Planeamento Orçamento

- ☒ Recpcionar via branca do PAD e arquivar juntamente com factura
- ☐ Nomear funcionário para acompanhar manutenção
- ☐ Acompanhar fornecedor na realização da manutenção
- ☒ Não aplicáveis

Elaborar Relatório

- ☐ Escrever

Não Aplicável	<input type="checkbox"/>
Nº PAD	<input type="text" value="123"/>
Data	<input type="text" value="25-05-2010"/>
Fornecedor	<input type="text" value="Moviflor"/>

Figura 4.18: Interface de edição de instâncias de processos

O utilizador alterando os campos (editáveis) que pretender na parte de edição de dados e carregando no botão "Alterar", a informação será enviada de imediato para o módulo Gestão das instâncias da camada lógica que, por sua vez, encaminhará a informação tratada para a camada de acesso a dados, para que a informação seja adicionada à base de dados.

Na parte inferior, parte de gestão do processo, poderá ser feita uma gestão da instância ao nível da sua execução. São assim apresentadas na *checklist* todas as tarefas e fases do processo, o que é uma referência para quem executar o processo. À medida que forem sendo executadas as tarefas, pode-se ir à *checklist* e assinalá-las como realizadas, para que haja um controlo daquilo que já foi efectuado no processo e o que ainda falta fazer. Contudo, podem ser inseridas algumas regras no processo o que faz com que algumas tarefas não possam ser marcadas como realizadas, sem que sejam realizadas previamente as suas tarefas precedentes. Isto acontece com a Tarefa "Nomear funcionário para acompanhar manutenção". Se o utilizador carregar na *checkbox* nesta

tarefa, para o processo da figura, não conseguirá alterar o estado da tarefa, uma vez que ainda existem tarefas precedentes a esta para realizar. O utilizador para saber o que precisa de fazer para desbloquear a tarefa, pode carregar no botão de informação que se encontra à frente da mesma, que será aberta a Popup da Figura 4.19, onde constará essa informação.



Figura 4.19: Popup com informação adicional

As tarefas podem ter outras tarefas como precedentes ou não, permitindo assim um grau de estruturação do processo variável. Contudo, se, no limite, forem definidas precedências entre todas as tarefas do processo estamos perante um processo de workflow.

As tarefas podem ainda ser marcadas como não aplicáveis, significando isto que não faz sentido a realização da tarefa em questão para a instância do processo a gerir. Neste caso, a tarefa é desloca para baixo da área "Não aplicáveis", para não confundir o utilizador e para que este se foque naquilo que precisa mesmo de realizar. Através da colocação de tarefas como não aplicáveis, podemos ter um modelo do processo e posteriormente podemos ter variantes desse modelo aquando a gestão de instâncias. Isto é importante porque, como vimos no Capítulo 3.3, existe uma grande diversidade de processos e a gestão de processos para o mesmo equipamento também pode variar de instância para instância. Assim, no modelo do processo podem estar

as tarefas todas que devem ser realizadas, e aquando a gestão de uma instância do processo o utilizador é que decide se a tarefa em questão se aplica ou não ao contexto em que o processo se está a desenrolar.

Ainda existe mais uma regra que pode ser colocada sob as tarefas. Assim, as tarefas podem ser obrigatórias ou não. As tarefas que são obrigatórias, o utilizador não pode colocá-las como não aplicáveis, estando assim a *checkbox* "Não aplicável" inactiva.

Quando o utilizador carrega na *checkbox* de uma tarefa, não precisa de carregar em nenhum botão para efectuar essa alteração no servidor, porque ela é executada por um pedido AJAX. Isto foi implementado desta forma, para facilitar a gestão do processo. Quando o utilizador marca como realizada uma tarefa, pode desbloquear outra tarefa, assim isto é feito pelo sistema em tempo real.

O utilizador pode ainda associar informação complementar às tarefas. Para isso tem de carregar na tarefa na qual pretende fazê-lo e, de seguida, aparecerá o menu da direita onde é possível gerir-se uma única tarefa. Na Figura 4.17 o menu encontra-se aberto para a tarefa seleccionada. É possível através do menu desta tarefa adicionar-se os campos "NºPAD", "Data" e "Fornecedor". Isto porque, estes atributos são os que estão definidos no modelo do processo para a tarefa em questão.

Capítulo 5

Gestor de conteúdos

Neste capítulo será efectuada uma abordagem ao módulo gestor de conteúdos, referido no Capítulo 3.3, desde a sua concepção até à sua implementação.

Começa-se na Secção 5.1 por mencionar os requisitos que este módulo deve satisfazer. De seguida, na Secção 5.2 são apresentados alguns aspectos de concepção deste módulo, nomeadamente a arquitectura geral, o modelo de dados e a interface idealizada. Por último, na Secção 5.3, é apresentada a interface final deste módulo e um exemplo do seu funcionamento.

5.1 Análise de requisitos

Como vimos no Capítulo 3, existem um conjunto de factores que dificultam a gestão dos conteúdos de processos semi-estruturados:

- a forma como é gerido um processo depende dos seus actores - dificulta a uniformização dos conteúdos;
- elevado número de actores externos intervenientes nos processos - torna obrigatório gerir-se conteúdos em diferentes suportes;
- conteúdos têm valor em diferentes contextos - exige flexibilidade na forma como são geridos os conteúdos.

Estes factores implicam (a um sistema de suporte) a capacidade de gerir conteúdos provenientes de diferentes suportes e flexibilidade para permitir o acesso aos conteúdos em diferentes contextos. Foi para dar resposta a isto, que surgiu o módulo *Gestor de conteúdos*. Antes da implementação deste módulo, foram identificados um conjunto de requisitos que este deve satisfazer para cumprir a função a que se destina (ver Tabela 5.1).

A gestão de conteúdos em formato de árvore é uma das formas mais intuitivas de mostrar ao utilizador as relações entre os conteúdos e organizá-los, daí que se tenha optado por apresentar os conteúdos em formato de árvore para este módulo (Req.1).

Nº	Requisito
1	Os conteúdos devem ser apresentados na forma de árvore
2	A árvore de conteúdos deve possuir nós que representem ficheiros relacionados com o processo
3	A árvore de conteúdos deve possuir nós que representem emails relacionados com o processo
4	Para cada modelo de processo a estrutura da árvore de conteúdos da instância deve ser a mesma
5	Carregando-se num nó da árvore deve-se abrir o conteúdo desse nó no suporte em que ele se encontrar

Tabela 5.1: Requisitos do módulo de gestão de conteúdos

Como os conteúdos podem estar em email e ficheiros de diferentes formatos, é necessário nós na árvore que representem esses conteúdos (Req.2 e Req.3).

A árvore de conteúdos a apresentar aquando a gestão dos conteúdos de uma instância do processo, deve possuir a mesma estrutura para todos as instâncias de processo pertencentes ao mesmo modelo (Req.4). Assim, juntamente com o modelo do processo, deve-se definir a estrutura dos conteúdos para o processo em questão

Os nós da árvore podem conter conteúdos de diferentes suportes. Para o utilizador não é importante o suporte em que se encontra a informação, o importante é a informação estar facilmente acessível quando ele a quiser aceder. Assim, o utilizador carregando num nó da árvore, deverá abrir o conteúdo independentemente do seu tipo (Req.5).

5.2 Conceção

Nesta secção, serão apresentados alguns aspectos da concepção deste módulo para que dê resposta aos requisitos mencionados na secção anterior. Assim, em primeiro lugar, será apresentada a arquitectura geral da aplicação. De seguida, dar-se-á mais ênfase ao modelo de dados da aplicação e às interfaces da mesma.

5.2.1 Arquitectura

A arquitectura deste módulo (Figura 5.1) foi dividida em três camadas (camada de acesso a dados, camada lógica e camada de apresentação) tal como aconteceu com o gestor de processos. No entanto, existem diferenças substanciais entre ambas: enquanto que no gestor de processo a única fonte de dados é a BD, neste módulo os dados podem ser provenientes de três fontes de dados:

- base de dados
- listas de SP

- sistema de ficheiros

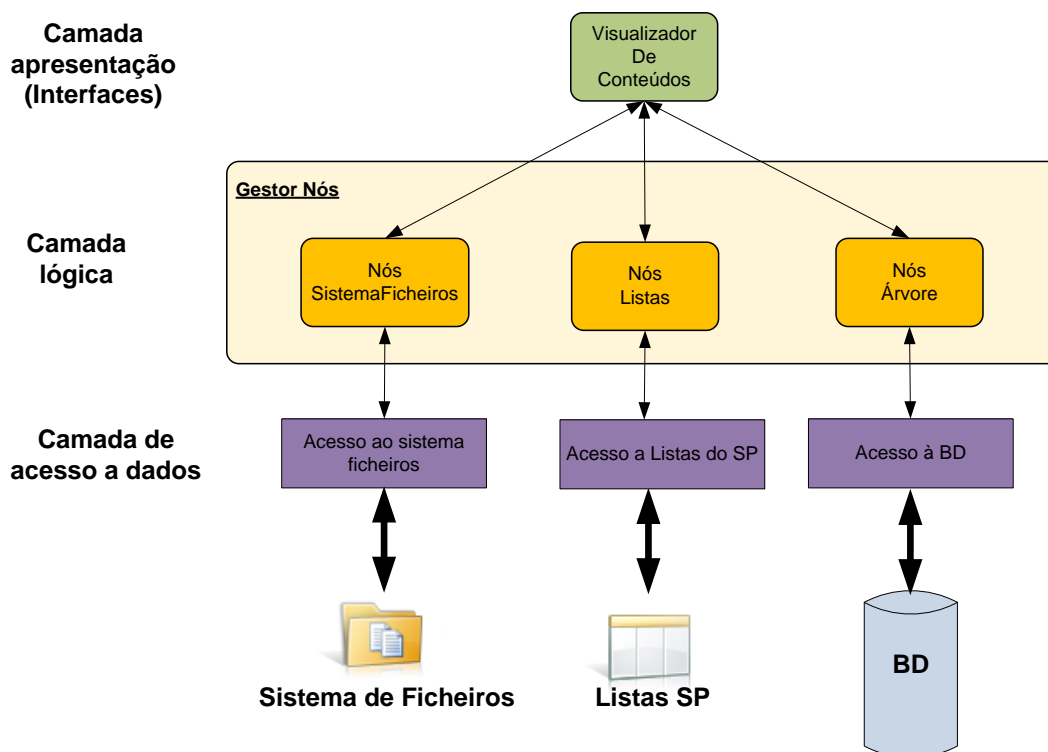


Figura 5.1: Arquitectura geral do gestor de conteúdos

Na base de dados ficará apenas guardada a estrutura da árvore de conteúdos para cada modelo de processo, assim como os parâmetros adicionais de cada nó da árvore (caminho da pasta, caminho do ficheiro, tipo de emails a mostrar, etc.).

Uma lista do SP é em tudo idêntica a uma tabela de uma base de dados, onde as colunas de uma lista correspondem às colunas da tabela, com a grande vantagem de poder armazenar vários tipos de dados, desde datas, documentos, hiperligações, imagens ou mesmo cálculos com base no valor de outras colunas, bastando para isso escolher o tipo de coluna e a sua configuração. As listas podem beneficiar das funcionalidades de correio electrónico, se a recepção ou envio de correio estiver activado no site. Neste caso, os emails recebidos são armazenados em listas. É assim necessário para este módulo aceder à informação contida em listas do SP para integrar a informação dos emails no gestor de conteúdos.

Alguns dos nós que estão na base de dados, poderão dizer respeito a pastas ou ficheiros existentes nos directórios da aplicação, logo é necessário que este módulo aceda a informação existente no sistema de ficheiros.

Para aceder a estes três fontes de dados, serão criadas três classes na camada de acesso a dados, sendo cada uma responsável por aceder a informação em diferentes suportes (listas, BD e sistema de ficheiros). Essas classes serão:

- **acesso a sistema de ficheiros** - classe responsável por aceder a um directório do site e retornar os elementos que a constituem para a camada superior;
- **acesso a listas em SP** - classe responsável por aceder a uma determinada lista do SharePoint e retornar essa informação para a camada superior;
- **acesso à BD** - classe responsável por aceder à base de dados e retornar a informação para a camada superior.

A informação retornada pelas classes da camada de acesso a dados, é tratada pelas classes da camada lógica antes de ser passada para a camada de apresentação. Por isso, na camada lógica existem três classes distintas, cada uma delas especializada em tratar a informação de um tipo de nó da árvore:

- **NósSistemaFicheiros** - classe responsável pelos nós da árvore que se encontram no sistema de ficheiros;
- **NósListas** - classe responsável pelos nós da árvore que se encontram em listas;
- **NósArvore** - classe responsável pelos nós da árvore que se encontram em BD.

Na camada de apresentação encontra-se apenas uma interface para possibilitar a visualização dos conteúdos de uma instância do processo. Não é necessário neste módulo uma interface para configurar os nós da árvore para um modelo de processo, porque isso deverá ser efectuado aquando a criação e/ou edição de modelos de processos. Mas como as interfaces do gestor de processos referentes à criação e edição de modelos não foram implementadas, a configuração da árvore de conteúdo terá de ser realizada, por enquanto, directamente na BD.

O objectivo da interface de visualização de conteúdos é permitir ao utilizador aceder aos conteúdos do processo através de uma árvore. Para isso, aquando a abertura da interface, deve ser preenchida a árvore com a estrutura inicial definida na base de dados para o processo em questão, tendo-se assim de início apenas nós provenientes da BD. Posteriormente, o fluxo de execução deste módulo, desde o ponto em que o utilizador carrega num nó, até visualizar os nós que se encontram a um nível inferior ao carregado, deve ser o seguinte:

1. Após o utilizador na interface carregar num nó para o expandir, a classe GestorNós da camada lógica avalia que tipo de nó se trata e reencaminha para a classe do tipo de nó correspondente.
2. A classe do tipo de nó que foi expandido, invoca a classe da camada de acesso a dados para lhe fornecer os nós filhos do nó que foi seleccionado pelo utilizador.
3. Essa informação é tratada pela classe da camada lógica e de seguida é enviada para a interface para que o utilizador veja o nó expandido.

5.2.2 Modelo de dados

O modelo de dados deste módulo é mais simples que o do gestor de processos. Ele é constituído por um único pacote de dados lógico denominado por **Árvore Conteúdos** (ver Figura 5.2). Este pacote é constituído pelas seguintes entidades:

- NoConteudo
- TipoNo

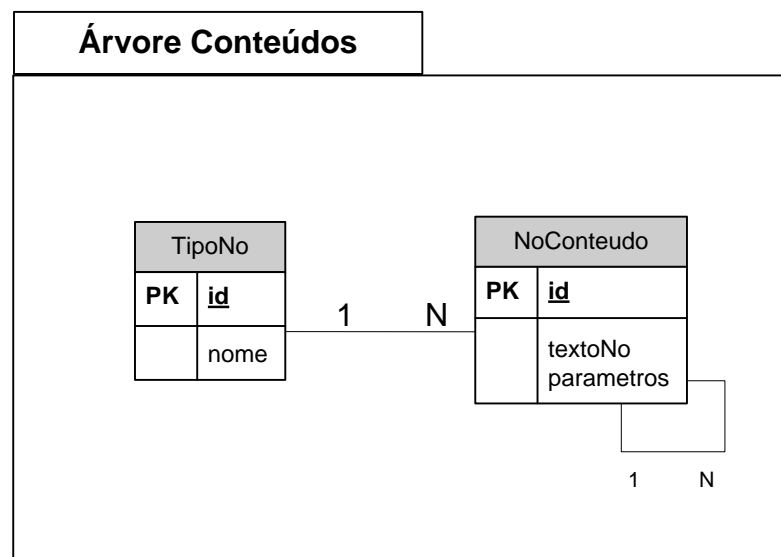


Figura 5.2: Modelo de dados do pacote árvore de conteúdos

A entidade NoConteudo representa os nós provenientes da BD que irão aparecer na árvore de conteúdos. Esses nós, terão um texto a mostrar ao utilizador, que será guardado no campo "TextoNo" e um valor oculto, que será guardado no campo "parametros" e servirá para guardar no nó informação que poderá ser útil caso o utilizador carregue nele posteriormente.

A entidade tipoNo representa os tipos de nós que a árvore de conteúdos irá possuir. Sabendo o tipo de um nó, o Gestor de Nós da camada lógica conseguirá encaminhar o tratamento de um nó a expandir, para a classe à qual ele pertence. Os tipos de nós que irá possuir a árvore de conteúdos, por enquanto, serão:

- **SubArvore:** Nó que terá os seus filhos na entidade NoConteudo. Serve para organizar conteúdos;
- **Pasta Processo:** Nó que terá os seus filhos numa pasta do sistema de ficheiros exclusiva para a instância do processo;
- **Pasta:** Nó que terá os seus filhos numa pasta do sistema de ficheiros;

- **Emails Processo:** Nó que terá os seus filhos numa lista onde se encontram os emails dos processos;
- **Emails Fornecedor:** Nó que terá os seus filhos numa lista onde se encontram os emails dos fornecedores.

Alguns tipos de nós precisam de informação adicional para obter os seus filhos. Essa informação é guardada na entidade NoConteudo no campo "parametros" . Os parâmetros necessários para cada tipo de nó são os seguintes:

- SubArvore: não precisa de parâmetro, porque o nó sendo do tipo subArvore já se sabe que os seus filhos estão na base de dados;
- Pasta Processo: não precisa de nenhum parâmetro porque basta saber o *id* do processo que está a ser gerido.
- Pasta: precisa do caminho da pasta no sistema de ficheiros para posteriormente aceder aos seus filhos.
- Emails Processo: não precisa de nenhum parâmetro porque basta saber o *id* do processo que está a ser gerido.
- Emails Fornecedor: precisa do *id* do fornecedor que se pretende listar os emails.

O tipo de no SubArvore implicou a relação de 1 para N entre os nós de conteúdos. Este tipo de nó serve para organizar conteúdos, podendo estes estar organizados hierarquicamente, daí esta relação.

O pacote Árvore Conteúdos interage com o pacote Modelo dos Processos para que os nós de conteúdos fiquem associados ao modelo ao qual pertencem. Tal como ilustra a Figura 5.3 um modelo de processo pode possuir vários nós de conteúdo. Na figura o pacote Modelo de Processos aparece apenas com uma entidade, porque é a que interage com o pacote Árvore Conteúdos, mas como vimos no Capítulo 4.2.2.1 este pacote possui mais entidades.

5.2.3 Interface

O esboço da interface idealizada para o utilizador visualizar a agregação de conteúdos, está presente na Figura 5.4.

Os conteúdos serão organizados em árvore, podendo o utilizador expandir um nó, caso pretenda obter os seus filhos (nós a um nível inferior), ou colapsar, caso pretenda ocultar os nós filhos.

A árvore de conteúdos para um processo, irá agregar conteúdos provenientes de diferentes suportes na mesma interface, sem que o utilizador perceba a origem dos mesmos. O importante para o utilizador é, quando precisar de um conteúdo, saber onde ele se encontra e estar facilmente acessível, independentemente da forma como ele será apresentado (pdf, excel, etc.).

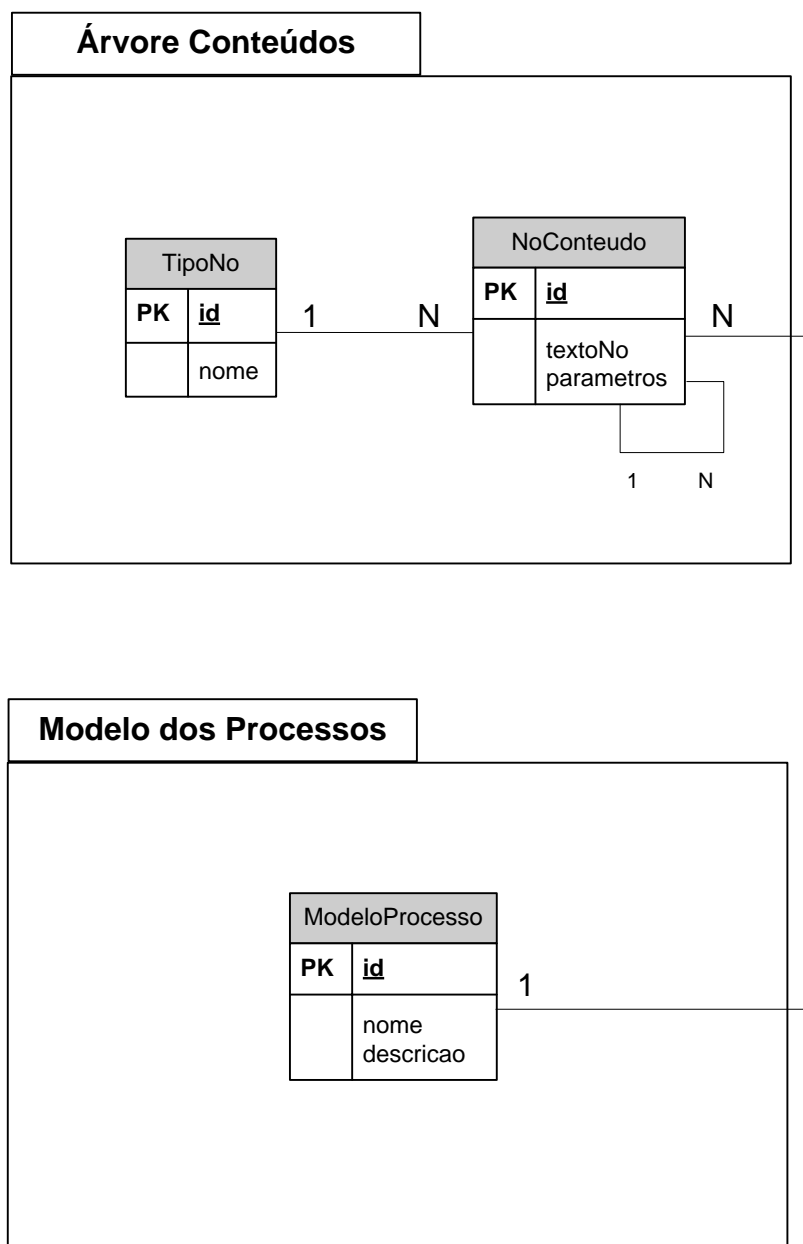


Figura 5.3: Modelo de dados do gestor de conteúdos

Um processo pode ter uma pasta com conteúdos que apenas lhe dizem respeito. O nó que deverá apresentar essa informação, deve ser do tipo Pasta Processo. O nó da interface com o nome "Pasta Processo" se for desse tipo, quando o utilizador o tentar expandir, serão apresentados os filhos que se encontram no sistema de ficheiros no directório da pasta do processo específico.

Já o nó "Certificados", se for um nó do tipo Pasta, quando o utilizador o expandir, os seus filhos serão obtidos através do sistema de ficheiros para o directório que se encontra oculto no valor do nó.

O nó "Emails", se for do tipo SubArvore, quando o utilizador o expandir os seus filhos serão

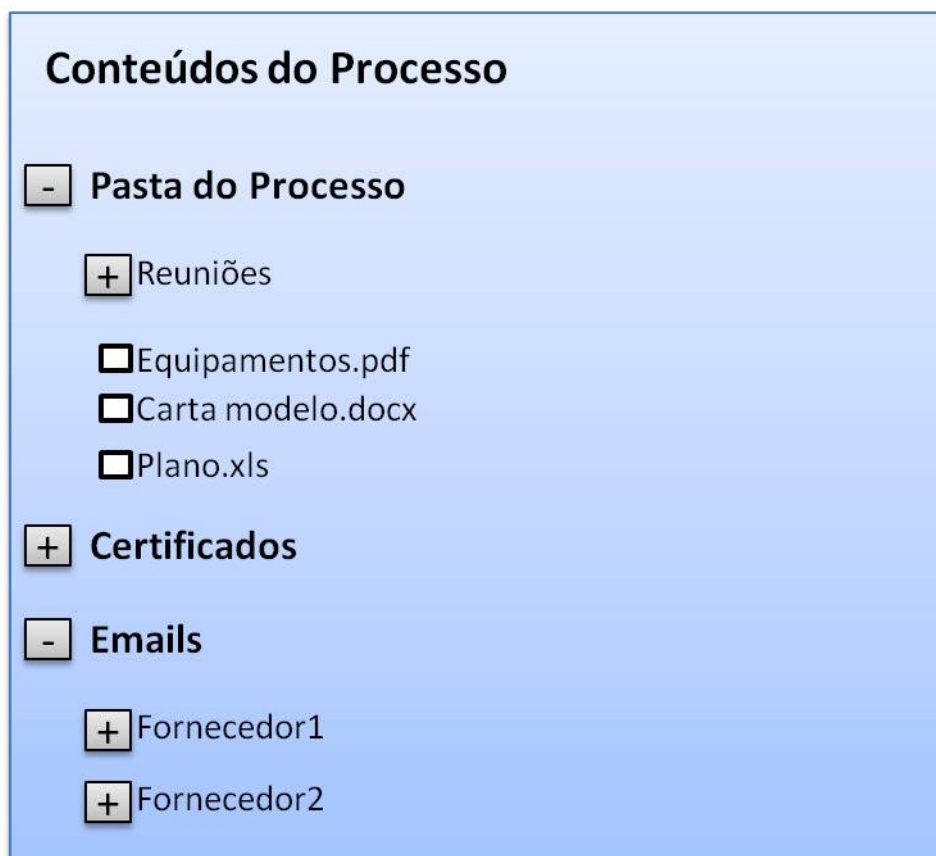


Figura 5.4: Esboço da interface do gestor de conteúdos

provenientes da base de dados. Neste caso, os seus filhos seriam "Fornecedor1" e "Fornecedor2". Se estes dois nós fossem do tipo Emails Fornecedor, quando o utilizador os expandir, deverá estar oculto no valor dos nós o *id* do fornecedor para o qual deveram aparecer os emails.

5.3 Implementação

Este módulo foi implementado com as mesmas tecnologias que o gestor de processos.

Para se perceber a forma como foi implementado este módulo, será dado um exemplo de um gestor de conteúdos criado para um processo. Assim, em primeiro lugar ilustrar-se-á os nós pertencentes a esse gestor ao nível da base de dados e posteriormente a forma como a árvore de agregação evolui com as acções efectuadas pelo utilizador na interface.

Na base de dados está a informação referente à estrutura da árvore de conteúdos a apresentar para cada modelo do processo. Como já foi dito anteriormente, esta informação deveria ser colocada na base de dados aquando a edição ou criação de um modelo de processo.

A tabela que contém todos os tipos de nós que uma árvore pode conter, está presente na Figura 5.5.

id	nome
1	Pasta Processo
2	SubArvore
3	Pasta
4	Emails Processo
5	Emails Fornecedor

Figura 5.5: Dados da tabela tipoNo

Por sua vez, na Figura 5.6, está presente a configuração de um gestor de conteúdos para um modelo de processo. Tal como está no modelo de dados, esta tabela deveria ter um atributo que era o modelo ao qual pertence o nó, e efectivamente tem, só não foi colocado na figura para simplificar, uma vez que aqueles nós pertencem todos ao mesmo modelo.

id	textoNo	parametros	tipoNo	noPai
1	Documentos do Processo		1	NULL
2	Certificados		2	NULL
10	Controlo	Manutenção/ESI/Controlo	3	2
11	Elevadores	Manutenção/ESI/E01 - Elevadores/Certificados	3	2
12	Ar Comprimido	Manutenção/ESI/E06 - Ar comprimido/Certificados	3	2
13	Emails		2	NULL
14	Processo		4	13
17	Fornecedores		2	13
18	Schneider	2	5	17
19	Efacec	1	5	17

Figura 5.6: Dados da tabela NoConteudo

Na tabela NoConteudo estão todos os nós que irão fazer parte da árvore de conteúdos a mostrar ao utilizador, as relações entre eles, os parâmetros dos nós, e o texto a mostrar ao utilizador.

A interface do gestor de conteúdos está presente na Figura 5.7. Analisando a interface, repara-se que a estrutura da árvore que é mostrada ao utilizador, está fiel à informação que está contida na tabela NoConteudo.

Através desta interface é possível agregar conteúdos de diferentes suportes e provenientes de diferentes contextos. Isso acontece com os Certificados, em que embora estejam organizados dentro dos directórios de cada área de actividade, através do gestor de conteúdos está-se a visualizar numa única interface os certificados que interessam para este modelo. A abordagem utilizada permite agregar na mesma interface conteúdos que estejam espalhados pelo sistema de ficheiros e listas de SP, definindo-se para isso na base de dados os conteúdos a visualizar e as suas proveniências.



Figura 5.7: Interface do gestor de conteúdos

Quando esta interface é iniciada, apenas aparecem os nós da tabela NoConteúdo que têm noPai igual a *Null*. À medida que o utilizador vai carregando nos nós para os expandir, vão sendo efectuados pedidos AJAX ao servidor. Os nós da árvore têm dois atributos, um é o texto a apresentar ao utilizador e outro o valor que fica oculto e que será passado para o servidor quando se carregar no nó. Com base nesse valor, quando se carrega num nó, o Gestor de Nós da camada lógica encaminha o nó para ser tratado pela classe correspondente ao tipo de nó que foi carregado. Posteriormente, esta classe obtém os filhos do nó carregado e envia-os para a camada superior.

Se o nó se tratar de um ficheiro ou um email, abre directamente o seu conteúdo independentemente do suporte em que encontrar. No fundo os conteúdos são agregados no mesmo local, para que estejam acessíveis ao utilizador e para que posteriormente possam ser acedidos em diferentes plataformas, isto é, tanto se pode carregar num pdf e abrir um leitor de pdf, como se pode carregar num ficheiro de excel e abrir o Excel, como carregar num email e abrir um formulário que mostra o email com base na informação que consta numa lista do SP.

A organização das pastas dos conteúdos, pode ser feita com recurso ao explorador do windows. Se se carregar na opção "Abrir" que se encontra à frente de alguns nós, abrirá o explorador, dando

ao utilizador a sensação de estar a gerir conteúdos que estão no seu computador. Na Figura 5.8 está o explorador do windows aberto para o nó Elevadores da árvore de conteúdos. Assim, é possível arrastar, apagar e organizar os documentos de uma forma flexível. Se o utilizador quiser passar do seu computador 20 certificados para a pasta dos Elevadores, só os têm de arrastar para o explorador que passarão para o servidor.

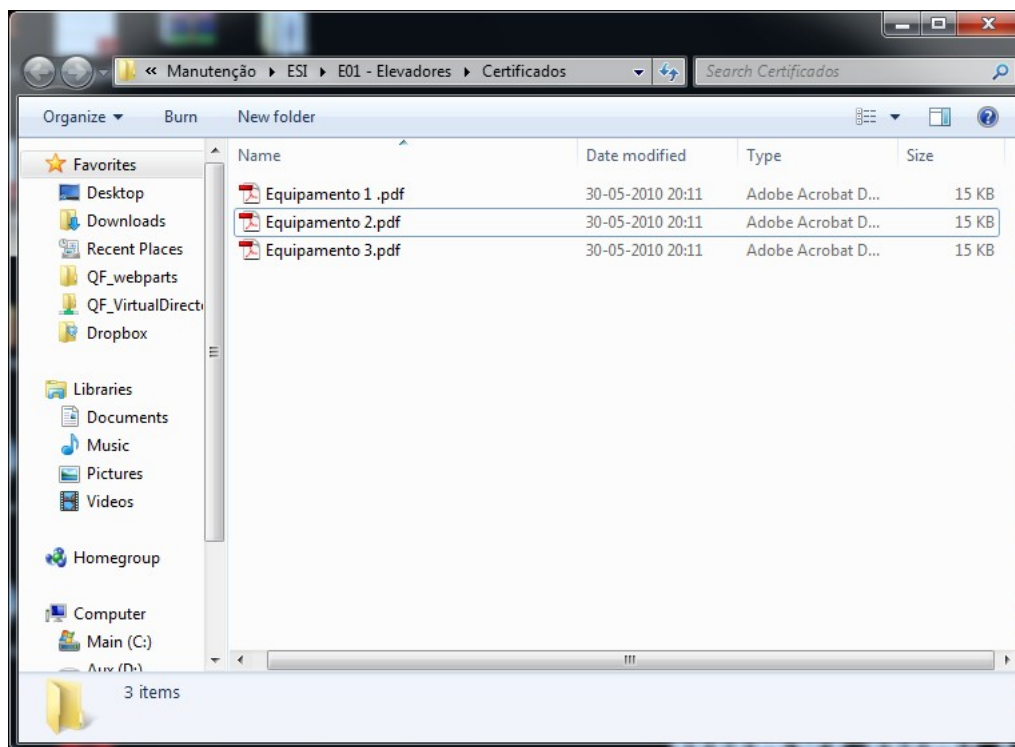


Figura 5.8: Explorador do windows para organização dos conteúdos

Nem todos os nós podem ser abertos com o explorador, por isso é que nem todos os nós da árvore da Figura 5.7 têm o link "Abrir" disponível. Só os nós do tipo "Pasta de Processo" ou "Pasta" é que permitem a abertura com o explorador. Isto acontece, porque os restantes nós não fazem parte do sistema de ficheiros, uns estão na base de dados os seus filhos (nós do tipo SubArvore) e outros estão em listas (nós do tipo "Emails Fornecedor" e "Emails Processo").

Capítulo 6

Conclusões e perspectivas de desenvolvimento

Os objectivos propostos para este projecto foram cumpridos. Começou-se por perceber e identificar as necessidades de um sistema de trabalho semi-estruturado ao nível do seu sistema de suporte, analisando-se para isso os STM da FEUP. Por sua vez, essas necessidades deram origem a requisitos para o sistema de suporte, e numa última fase foram desenvolvidos dois módulos para dar suporte a sistemas de trabalho semi-estruturados:

- gestor de processos;
- gestor de conteúdos;

O gestor de processos, com recurso a uma *checklist*, permite de uma forma flexível a gestão de processos com diferentes níveis de estruturação, dentro de um conjunto de regras. Para além disso, mantém a capacidade de consolidação de conteúdos, tornando assim possível ter-se uma visão consolidada do estado de uma área de trabalho.

O gestor de conteúdos tem a capacidade de agregar na mesma interface conteúdos provenientes de diferentes suportes e originados em diferentes contextos, mostrando ao utilizador esses mesmos conteúdos numa árvore hierárquica, que agrega na mesma interface todos os conteúdos relacionados com o processo. Embora os conteúdos sejam agregados, quando o utilizador pretende aceder aos mesmos terá obrigatoriamente de utilizar ferramentas diferentes, porque está a aceder a conteúdos de natureza distinta (pdf, docx, etc.). Uma das maiores potencialidades deste módulo, está na possibilidade de se gerir conteúdos através do explorar do Windows, o que torna bastante flexível a organização de conteúdos, principalmente quando se pretende organizar vários conteúdos em simultâneo.

Após a implementação dos módulos procedeu-se a uma integração dos mesmos como componentes (Peças Web) no sistema de suporte aos STM. Foram realizados testes, relativos à execução dos módulos, que receberam opiniões muito favoráveis. Contudo, não foi possível realizar-se uma avaliação dos resultados, por falta de tempo, uma vez que a parte de análise do sistema de trabalho e de implementação dos módulos foi demorada.

Dado aquilo que foi efectuado, é possível perspectivar um conjunto de funcionalidades que podem ser acrescentadas a ambos os módulos, após ser realizada a avaliação dos mesmos.

No gestor de processos poderão ser acrescentadas as seguintes funcionalidades:

- atribuição de uma tarefa de um processo a um actor ou conjunto de actores;
- envio de alertas por email, popup, ou SMS a avisar os actores da necessidade de realização de tarefas;
- permitir a criação de planos anuais de manutenção, para que no início do ano civil seja possível realizar-se o plano do ano e posteriormente seja gerada automaticamente uma instância do processo para cada item do plano;
- quando um processo é concluído, encaminhar alguns conteúdos do processo para dar seguimento aos mesmos, p.ex., factura que lhe está associada poderia ser reencaminhada automaticamente para quem trata das despesas e dá ordem de pagamentos das facturas.
- permitir a colocação de instruções de trabalho aquando a criação do modelo do processo, para que o utilizador possua mais informações de auxílio à execução.
- dar a possibilidade de o utilizador acrescentar tarefas a uma instância do processo. Isto seria útil na gestão de processos semi-estruturados, em que muitas vezes vão surgindo tarefas novas ao longo da execução do processo.

Ao nível do gestor de conteúdos, podem ser acrescentadas as seguintes funcionalidades:

- tornar o componente de gestão de conteúdos, um componente genérico de conteúdos, que possa agregar e apresentar ao utilizador conteúdos relacionados com outras entidades, como p.ex., agregar conteúdos associados a uma unidade da organização. Desta forma, este módulo, seria um gestor de conteúdos flexível que serviria para diferentes contextos, para além do contexto do processo.
- acrescentar novos tipos de nós na árvore de conteúdos.

Anexo A

Lista de equipamentos, sistemas e infraestruturas dos STM

Equipamento, sistema ou infraestrutura	Tipo Manutenção		
	MRI	MCRE	MPRE
Postos de transformação:			
• Transformadores		X	X
• Equipamento de corte e protecção		X	X
• Equipamento de comando		X	X
Rede de baixa tensão:			
• Quadros eléctricos de distribuição	X		
• Cablagem	X		
• Contadores de energia	X		
• Compensadores de factor de potência	X		
• Filtros de distorção harmónica	X	X	

Tabela A.1: Equipamentos, sistemas e infraestruturas no âmbito das instalações eléctricas

Equipamento, sistema ou infraestrutura	Tipo Manutenção		
	MRI	MCRE	MPRE
Inversores de rede		X	
UPS		X	X
Grupos geradores:			
• Motor diesel		X	X
• Alternador		X	X
• Quadros de comando		X	X

Tabela A.2: Equipamentos, sistemas e infraestruturas no âmbito dos sistemas de alimentação de emergência

Equipamento, sistema ou infraestrutura	Tipo Manutenção		
	MRI	MCRE	MPRE
Infra-estrutura geral:			
• Cablagem	X		
• Controladores locais		X	X
• Controladores intermédios		X	X
• Unidade central		X	X
Rede detecção de incêndios e alarme:			
• Cablagem	X		
• Detectores	X		X
• Dispositivos de alarme	X		
• Central de controlo		X	
Controlo de acessos:			
• Cablagem	X		
• Equipamentos terminais	X		
• Portas rotativas		X	
• Central de controlo	X		

Tabela A.3: Equipamentos, sistemas e infraestruturas no âmbito da gestão técnica centralizada

Equipamento, sistema ou infraestrutura	Tipo Manutenção		
	MRI	MCRE	MPRE
Aquecimento:			
• Caldeiras a gás		X	X
• Radiadores	X		
• Convectores	X		
• Cilindros eléctricos		X	X
Ventilação:			
• Ventilação/extracção sanitários		X	X
• Extracção salas aula		X	X
• Desenfumagem		X	X
• UTA's		X	X
Ar condicionado:			
• Chillers		X	X
• Torres de refrigeração		X	X
• Ventilo-convectores		X	X
• Splits		X	X
• Rooftops		X	X
• UTA's		X	X

Tabela A.4: Equipamentos, sistemas e infraestruturas no âmbito do AVAC

Equipamento, sistema ou infraestrutura	Tipo Manutenção		
	MRI	MCRE	MPRE
Bombas de água:			
• Bombas de drenagem		X	X
• Bombas do abastecimento de água potável		X	X
• Bombas do abastecimento de água não potável		X	X
• Bombas do sistema de combate a incêndios		X	X
• Bombas sistema de rega		X	X
• Bombas próprias do Labº de Hidráulica		X	X
• Bombas doseadoras de hipoclorito		X	
Equipamentos de segurança:			
• Extintores	X	X	
• Carreteis	X	X	
• Sinalética de segurança	X		
• Plantas de emergência	X		
• Mantas ignífugas	X		
• Sistema fixo de extinção	X	X	
• Malas e caixas de 1ºs socorros	X		
Elevadores:			
• Elevadores eléctricos		X	X
• Elevadores hidráulicos		X	X
• Monta-livros da Biblioteca		X	X
Sistema de alarme anti-intrusão		X	
Iluminação interior:			
• Equipamento de comando	X		
• Sensores de presença	X		
• Armaduras de diferentes tipos	X		
Iluminação exterior:			
• Equipamento de comando	X		
• Armaduras de diferentes tipos	X		
Barreiras dos parques:			
• Barreiras		X	X
• Leitores de cartões de identificação		X	X
• Sistema de controlo de abertura		X	X
• Semáforos indicativos da existência de vagas		X	X
• Comunicações áudio entre barreiras e central		X	X
Rede de ar comprimido		X	X
Rede de gás combustível		X	
Rede telefónica:			
• Central fixa		X	X
• Equipamento de comunicação com redes móveis	X		
• Telefones		X	X
Rede estruturada:			
• Cablagem	X		
• Bastidores	X		
Rede de gases especiais p/ uso laboratorial		X	
Detecção de presença de hidrogénio		X	X

Tabela A.5: Outros equipamentos, sistemas e infraestruturas

Referências

- [1] M. Pesic, H. Schonenberg, e W. M. P. van der Aalst. Declare: Full support for loosely-structured processes. *Enterprise Distributed Object Computing Conference, 2007. EDOC 2007. 11th IEEE International*, página 287, 2007.
- [2] Charlie Calvert e Dinesh Kulkarni. *Essential LINQ*. Addison-Wesley Professional, 2009.
- [3] Matthew MacDonald e Mario Szpuszta. *Pro ASP.NET 3.5 in C# 2008, Second Edition*. Apress, 2008.
- [4] Bill Evjen, Scott Hanselman, Farhan Muhammad, S. Srinivasa Sivakumar, e Devin Rader. *Professional ASP.NET 2.0*. Wrox Press Ltd., Birmingham, UK, UK, 2005.
- [5] Bill Evjen, Scott Hanselman, e Devin Rader. *Professional ASP.NET 3.5 In C# and VB*. Wrox Press Ltd., Birmingham, UK, UK, 2008.
- [6] Matthew MacDonald. *Beginning ASP.NET 3.5 in C# 2008: From Novice to Professional, Second Edition (Beginning: from Novice to Professional)*. Apress, Berkely, CA, USA, 2007.
- [7] Robin Dewson. *Beginning SQL Server 2005 for Developers: From Novice to Professional (Expert's Voice)*. Apress, Berkely, CA, USA, 2006.
- [8] Microsoft Corporation. *7 Development Projects for Office SharePoint Server 2007 and Windows SharePoint Services*. Microsoft Press, 2006.
- [9] Scot P. Hillier. *Microsoft SharePoint: Building Office 2007 Solutions in C#2005*. Apress, Berkely, CA, USA, 2007.
- [10] Tomas Andersson, Annika Andersson Ceder, e Ilia Bider. State-flow technique for business process analysis: case studies. *Logistics Information Management*, 15:34 – 45, 2002.
- [11] Alec Sharp e Patrick Mcdermott. *Workflow Modeling: Tools for Process Improvement and Application Development*. Artech House Publishers, 2001.
- [12] A.Bernstein e C.P.Schucan. Document and process transformation during the product life-cycle. *Information and Process Integration in Enterprises - Rethinking Documents*, T.Wakayame, S.Kannapan, C. M.Khoong, S.Navathe, and J.Yates, Eds.Norwell, MA:Kluwer Academic Publishers, 1998.
- [13] Charles Hill, Robert Yates, Carol Jones, e Sandra L. Kogan. Beyond predictable workflows: Enhancing productivity in artful business processes. *IBM Systems Journal*, 45(4):663–682, 2006.

- [14] A. Bernstein C.P. Schucan. How can cooperative work tools support dynamic group process? bridging the specificity frontier. Em *CSCW '00: Proceedings of the 2000 ACM conference on Computer supported cooperative work*, páginas 279–288, New York, NY, USA, 2000. ACM.
- [15] I. Bider e M. Khomyakov. Is it possible to make workflow management systems flexible? dynamical systems approach to business processes. *Groupware, International Workshop on*, 0:138, 2000.
- [16] Thomas P. Moran, Alex Cozzi, e Stephen P. Farrell. Unified activity management: supporting people in e-business. *Commun. ACM*, 48(12):67–70, 2005.
- [17] Michael Hammer e James Champy. *Reengineering the Corporation: A Manifesto for Business Revolution*. HarperBusiness, April 1994.
- [18] Wil M. P. van der Aalst, Maja Pesic, e Helen Schonenberg. Declarative workflows: Balancing between flexibility and support. *Computer Science - R&D*, 23(2):99–113, 2009.
- [19] M. Pesic e W. van der Aalst. A declarative approach for flexible business processes management. *Business Process Management Workshops*, páginas 169–180, 2006.
- [20] Steven Alter. *The Work System Method: Connecting People, Processes, and IT for Business Results*. Work System Press, 2006.